วิธีการหาเส้นทางโดยการใช้ข้อมูลตำแหน่งช่องสัญญาณเวลา

ROUTING ALGORITHM FOR FIXED SLOT ALLOCATION NETWORK

อัจฉรา แก้วเทพ

AJCHARA KAEWTHEP

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต

สาขาวิชาวิศวกรรมไฟฟ้า

บัณฑิตวิทยาลัย

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ. 2547

ISBN 974-15-1237-0

วิธีการหาเส้นทางโดยการใช้ข้อมูลตำแหน่งช่องสัญญาณเวลา

ROUTING ALGORITHM FOR FIXED SLOT ALLOCATION NETWORK

อัจฉรา แก้วเทพ
AJCHARA KAEWTHEP

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชาวิศวกรรมไฟฟ้า
บัณฑิตวิทยาลัย
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
พ.ศ. 2547

ISBN 974-15-1237-6

ROUTING ALGORITHM FOR FIXED SLOT ALLOCATION NETWORK

AJCHARA KAEWTHEP

A THESIS SUBMITTED IN PARTIAL FULFILLMENT

OF THE REQUIREMENT FOR THE DEGREE OF

MASTER OF ENGINEERING IN ELECTRICAL ENGINEERING

SCHOOL OF GRADUATE STUDIES

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

2004

ISBN 974-15-1237-6

COPYRIGHT 2004
SCHOOL OF GRADUATE STUDIES
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

หัวข้อวิทยานิพนธ์	วิธีการหาเส้นทางโดยการใช้ข้อมูลตำแหน่งช่องสัญญาณเวลา
นักศึกษา	นางสาวอัจฉรา แก้วเทพ
รหัสนักศึกษา	43061101
ปริญญา	วิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชา	วิศวกรรมไฟฟ้า
พ.ศ.	2547
อาจารย์ผู้ควบคุมวิทยานิพนธ์	รศ. ดร. ปิติเขต สู้รักษา

บทคัดย่อ

วิธีการหาเส้นทางในปัจจุบันมีหลายวิธี ซึ่งแต่ละวิธีออกแบบมาให้เหมาะสมกับโครงข่ายระบบสื่อสารพื้นฐานที่แตกต่างกัน วิทยานิพนธ์ฉบับนี้ได้นำเสนอวิธีการหาเส้นทางสำหรับระบบสื่อสารที่ใช้การจองช่องสัญญาณเวลาแบบตายตัวในเฟรมขนาดคงที่ วิธีการหาเส้นทางนี้จะนำเส้นทางที่มีความน่าจะเป็นในการสร้างเส้นทางสำเร็จจำนวนหนึ่ง ซึ่งคำนวณโดยใช้อัลกอริทึมการหาเส้นทางที่สั้นที่สุดมาทำการพิจารณาข้อมูลสถานะการใช้งานตำแหน่งช่องสัญญาณเวลาของโหนดทั้งหมดในเส้นทาง เพื่อเลือกเส้นทางที่เหมาะสมกับความต้องการของผู้ใช้ที่สุด

ในงานวิจัยได้พัฒนาวิธีการหาเส้นทางโดยการใช้ข้อมูลตำแหน่งช่องสัญญาณเวลานี้ลงบนโปรแกรมจำลองการทำงานระบบเครือข่าย OPNET เวอร์ชัน 6.0 และได้นำวิธีการดังกล่าวนี้ไปจำลองการทำงานบนโพรโตคอลสื่อสาร W-TDM ซึ่งสนับสนุนการสื่อสารที่กำหนดช่องสัญญาณเวลาแบบตายตัวในเฟรมขนาดคงที่ และผลที่ได้พบว่าอัตราการเกิดความผิดพลาดในการเลือกเส้นทาง จะน้อยกว่าวิธีการกำหนดเส้นทางโดยไม่ใช้ข้อมูลตำแหน่งช่องสัญญาณเวลาได้อย่างมีนัยสำคัญ

Thesis Title	Routing Algorithm for Fixed Slot Allocation Network
Student	Miss Ajchara Kaewthep
Student ID.	43061101
Degree	Master of Engineering
Programme	Electrical Engineering
Year	2004
Thesis Advisor	Assoc. Prof. Dr. Pitikhate Sooraksa

ABSTRACT

Nowadays, there are various routing algorithms which are appropriately designed for different communication networks. This thesis proposes a routing algorithm for fixed slot allocation within fixed frame size communication system. The algorithm utilizes Slot Usage Bit-Map (SUBM) of every node along a set of routes, which is the output of Shortest Path First algorithm, as to select the most appropriated route that meet the user's need.

The routing algorithm is designed and developed with OPNET 6.0, the network simulator program. The algorithm has been tested on W-TDM protocol, which is a communication protocol that uses fixed slot allocation onto fixed frame size technique. The result of the experiment shows that the proposed routing algorithm significantly reduces call block ratio better than none SUBM algorithm.

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สามารถสำเร็จลุล่วงไปได้ด้วยดีด้วยความกรุณาจากอาจารย์ที่ปรึกษา รศ. ดร. ปิติเขต สู้รักษา และ ผศ. มยุรี เลิศเวชกุล ที่ชี้แนะแนวทาง ให้วิชาความรู้และคำแนะนำในการแก้ไขปัญหาด้านต่าง ๆ รวมทั้งให้กำลังใจแก่ข้าพเจ้าเสมอมา ข้าพเจ้าขอกราบขอบพระคุณในความดีงามของท่านเป็นอย่างสูง

ขอขอบคุณสมาชิกทุกคนในครอบครัวที่คอยเป็นกำลังใจ ให้คำแนะนำ ให้ความช่วยเหลือและสนับสนุนในทุกด้านเสมอมา

ขอขอบคุณเพื่อน ๆ ในทีม รวมทั้งพี่ ๆ เพื่อน ๆ และน้อง ๆ ทุกคนที่ได้มีส่วนเกี่ยวข้องในการทำวิทยานิพนธ์ฉบับนี้ ซึ่งเป็นกำลังใจ และสร้างรอยยิ้มให้แก่ข้าพเจ้าอย่างสม่ำเสมอ

ขอขอบคุณภาควิชาวิศวกรรมสารสนเทศที่เอื้อเฟื้อสถานที่และอุปกรณ์ในการทำงานวิจัยจนสำเร็จลุล่วง

คุณค่าและคุณประโยชน์อันเกิดจากวิทยานิพนธ์ฉบับนี้ ข้าพเจ้าขอมอบให้แก่บิดา มารดา และผู้มีพระคุณแก่ข้าพเจ้าทุกคน

อัจฉรา แก้วเทพ

สารบัญ

หน้า

บทคัดย่อภาษาไทย.....I

บทคัดย่อภาษาอังกฤษ.....II

กิตติกรรมประกาศ.....III

สารบัญ.....IV

สารบัญตาราง.....VII

สารบัญรูป.....VIII

บทที่ 1 บทนำ.....1

1.1 ความเป็นมา.....1

1.2 ความสำคัญของปัญหา.....3

1.3 วัตถุประสงค์ของวิทยานิพนธ์.....4

1.4 ทฤษฎีและหลักการที่ใช้ในวิทยานิพนธ์.....5

1.4.1 วิธีการหาเส้นทาง.....5

1.4.2 หลักการของ W-TDM.....6

1.5 ของเขตของวิทยานิพนธ์.....6

1.6 ขั้นตอนการทำวิทยานิพนธ์.....8

1.7 โครงสร้างของวิทยานิพนธ์.....9

บทที่ 2 ทฤษฎีและหลักการที่เกี่ยวข้อง.....10

2.1 อัลกอริทึมการค้นหาเส้นทาง.....10

2.1.1 การหาเส้นทางแบบไม่มีการปรับตัว.....13

2.1.2 การหาเส้นทางแบบมีการปรับตัว.....14

2.1.3 การหาเส้นทางแบบเวกเตอร์ระยะทาง.....16

2.1.4 การหาเส้นทางแบบพิจารณาสถานะการใช้งานเส้นทางเชื่อมโยง.....19

2.2 การค้นหาเส้นทางกับการรับประกันคุณภาพการให้บริการ.....21

2.3 หลักการ W-TDM (Window-based TDM-like Scheduling).....25

บทที่ 3 การพัฒนาอัลกอริทึมการหาเส้นทางโดยใช้ข้อมูลตำแหน่งช่องสัญญาณเวลา.....28

3.1 นำเสนออัลกอริทึม.....28

สารบัญ (ต่อ)

หน้า

3.1.1 การจัดการเฟรมและพิจารณาเวลาหน่วง....29

3.1.2 การพัฒนาวิธีการหาเส้นทางโดยใช้ข้อมูลตำแหน่งช่องสัญญาณเวลา....34

3.2 การออกแบบและพัฒนาอัลกอริทึมลงในอุปกรณ์เครือข่ายจำลอง....42

3.2.1 อุปกรณ์สื่อสารปลายทางจำลอง....43

3.2.1.1 มอดูล trans (trans module)....44

3.2.1.2 มอดูล end (end module)....45

3.2.1.3 มอดูล sw (sw module)....51

3.2.2 อุปกรณ์เราท์เตอร์และสวิตช์จำลอง....52

3.2.2.1 มอดูล trans (trans module)....52

3.2.2.2 มอดูล mgmt (mgmt module)....52

3.2.2.3 มอดูล sw (sw module)....73

บทที่ 4 การจำลองและผลการจำลอง....79

4.1 กระบวนการจำลองระบบเครือข่าย....79

4.1.1 ข้อกำหนดของอุปกรณ์เครือข่ายจำลอง....79

4.1.2 โครงสร้างของระบบเครือข่ายจำลอง....80

4.1.3 การจำลอง....80

4.2 ผลการจำลอง....82

4.3 สรุปและวิเคราะห์ผลการจำลอง....85

บทที่ 5 สรุปและนำเสนอแนวทางในการทำวิจัย....86

5.1 บทสรุป....86

5.2 แนวทางในการวิจัยและพัฒนา....87

เอกสารอ้างอิง....88

ภาคผนวก....91

สารบัญ (ต่อ)

หน้า

ภาคผนวก ก 92

ประวัติผู้เขียน 98

สารบัญตาราง

ตารางที่ หน้า

3.1 ตารางเก็บข้อมูลการเชื่อมโยงกับโหนดข้างเคียง.....54

3.2 ตารางเส้นทาง.....55

3.3 ตารางจัดลำดับแพ็กเกตเพื่อส่งออก.....56

3.4 ตารางจัดลำดับแพ็กเกตเพื่อส่งออก.....56

3.5 ตารางข้อมูลแพ็กเกตแจ้งสถานะการใช้งานเส้นทางเชื่อมโยง.....57

3.6 ตารางข้อมูลแพ็กเกตแจ้งสถานะการใช้งานเส้นทางเชื่อมโยง.....63

สารบัญรูป

รูปที่ หน้า

2.1 อัลกอริทึมการหาเส้นทาง....13

2.2 การหาเส้นทางแบบเวกเตอร์ระยะทาง....17

2.3 การหาเส้นทางแบบเวกเตอร์ระยะทางกรณีมีเส้นทางเสียหายเกิดการคำนวณไม่สิ้นสุด....17

2.4 ตัวอย่างวิธีการ Split-Horizon....18

2.5 โครงสร้างของสวิตช์ในหลักการของ W-TDM....25

3.1 การพิจารณาเวลาหน่วงควบคุมและตำแหน่งเยื้องที่ทำให้เกิดการซิงโครนัสในเส้นทางจากโหนด i ไปยังโหนด n....33

3.2 การพิจารณาเวลาหน่วงเมื่อไม่สามารถจองในตำแหน่งซ้ายสุดของวินโดว์....33

3.3 การเทียบบิตโดยการ AND ชุดบิตข้อมูลในเส้นทางจากโหนด i ไปยังโหนด n....37

3.4 วิธีการเทียบชุดข้อมูลบิตโดยทำการ AND เป็นคู่....41

3.5 แบบจำลองภายในอุปกรณ์สื่อสารปลายทาง (End node model)....43

3.6 แผนผังสถานะภายในมอดูล trans (state diagram of trans module)....44

3.7 แผนผังสถานะของมอดูล end (state diagram of end module)....46

3.8 แผนผังสถานะของกระบวนย่อยในการสร้างแพ็กเกตร้องขอเพื่อเชื่อมโยงเส้นทาง....47

3.9 โครงสร้างของแพ็กเกตร้องขอเพื่อเชื่อมโยงเส้นทาง....47

3.10 แผนผังสถานะของกระบวนย่อยในการสร้างแพ็กเกตข้อมูล....50

3.11 โครงสร้างแพ็กเกตข้อมูล....51

3.12 แผนผังสถานะภายในมอดูล sw....51

3.13 แบบจำลองภายในอุปกรณ์เราท์เตอร์และสวิตช์ (Router and switch node model)....52

3.14 แผนผังสถานะภายในมอดูล mgmt....53

3.14 แผนผังสถานะภายในมอดูล mgmt....53

3.15 โครงสร้างแพ็กเกตแจ้งสถานะการใช้งานเส้นทางเชื่อมโยง....56

3.16 โฟล์วชาร์ตปรับปรุงข้อมูลเมื่อรับแพ็กเกตแจ้งสถานะการใช้งานเส้นทางเชื่อมโยง....58

3.17 โฟล์วชาร์ตปรับปรุงข้อมูลเมื่อรับแพ็กเกตแจ้งสถานะการใช้งานเส้นทางเชื่อมโยง (ต่อ)....59

3.18 โฟล์วชาร์ตฟังก์ชันการค้นหาเส้นทาง....60

3.19 โฟล์วชาร์ตฟังก์ชันการค้นหาเส้นทาง (ต่อ)....61

3.20 โฟล์วชาร์ตฟังก์ชันการค้นหาเส้นทาง (ต่อ)....62

3.21 โฟล์วชาร์ตฟังก์ชันการค้นหาเส้นทาง (ต่อ)....63

สารบัญรูป (ต่อ)

รูปที่ หน้า

3.22 ตัวอย่างการเก็บข้อมูลสถานะการใช้งานช่องสัญสัญญาณเวลาและตำแหน่งเยื้อง
ในการเทียบบิตแต่ละรอบ....64
3.23 โฟล์วชาร์ตฟังก์ชันการค้นหาเส้นทาง (ต่อ)....65
3.24 โฟล์วชาร์ตฟังก์ชันการค้นหาเส้นทาง (ต่อ)....66
3.25 โฟล์วชาร์ตฟังก์ชันการค้นหาเส้นทาง (ต่อ)....67
3.26 โฟล์วชาร์ตการทำงานของสถานะ call ในอุปกรณ์เราท์เตอร์จำลอง....70
3.27 โฟล์วชาร์ตการทำงานของสถานะ call ในอุปกรณ์เราท์เตอร์จำลอง (ต่อ)....71
3.28 โฟล์วชาร์ตการทำงานของสถานะ call ในอุปกรณ์สวิตช์จำลอง....72
3.29 โฟล์วชาร์ตการทำงานของจำลองกรณีที่ได้รับแพ็กเกตตอบรับแบบปฏิเสธ....73
3.30 โฟล์วชาร์ตสถานะภายในมอดูลสวิตช์....74
3.31 โฟล์วชาร์ตการทำงานในมอดูลสวิตช์ที่อุปกรณ์เราท์เตอร์จำลอง....77
3.32 โฟล์วชาร์ตการทำงานในมอดูลสวิตช์ที่อุปกรณ์สวิตช์จำลอง....78
4.1 ระบบเครือข่ายจำลองที่ใช้ในการทดลอง....79
4.2 เปรียบเทียบอัตราการปฏิเสธการร้องขอเพื่อเชื่อมโยงเส้นทางระหว่าง
กรณีที่ใช้วิธีการ SUBM และกรณีที่ไม่ได้ใช้วิธีการ SUBM....82
4.3 เปรียบเทียบอัตราการใช้งานช่องสัญญาเวลาในระบบระหว่าง
กรณีที่ใช้วิธีการ SUBM และกรณีที่ไม่ได้ใช้วิธีการ SUBM....83
4.4 เปรียบเทียบขอบเขตของเวลาหน่วงระหว่างกรณีที่ใช้วิธีการ SUBM
และกรณีที่ไม่ได้ใช้วิธีการ SUBM....84
4.5 เปรียบเทียบอัตราการสูญหายของแพ็กเกตระหว่างกรณีที่ใช้วิธีการ SUBM
และกรณีที่ไม่ได้ใช้วิธีการ SUBM....84

บทที่ 1
บทนำ

1.1 ความเป็นมา

ปัจจุบันการให้บริการในระบบสื่อสารข้อมูลแบบแพ็กเกตสวิตชิ่งนั้นมีการพัฒนารูปแบบของการให้บริการที่หลากหลายและแตกต่างกันไปตามการประยุกต์การใช้งานของผู้ใช้ ทิศทางการพัฒนาเป็นไปเพื่อรับประกันคุณภาพการให้บริการ (Quality of service, QoS) ของการสื่อสารให้เหมาะสมกับความต้องการของผู้ใช้ การรับประกันคุณภาพการให้บริการสามารถทำได้ในหลายกระบวนการ เช่น กระบวนการค้นหาเส้นทาง (routing process) การร้องขอเส้นทางเชื่อมโยงและอนุมัติเส้นทาง (call-setup and call admission control) การควบคุมปริมาณทราฟฟิก (traffic shaping) การตรวจวัด ทราฟฟิกและกำจัดแพ็กเกตที่ผิดเงื่อนไขข้อตกลงในการสื่อสารเป็นต้น [1] การค้นหาและสร้างเส้นทางนั้นเป็นกระบวนการแรกที่สามารถรับประกันคุณภาพการให้บริการได้โดยเลือกเส้นทางที่ทรัพยากรเพียงพอและสามารถรับประกันการให้บริการตามความต้องการของทราฟฟิกที่ขอใช้บริการได้ในปัจจุบันวิธีการหาเส้นทางมีอยู่หลายวิธี และแต่ละวิธีจะพิจารณาเลือกเส้นทางโดยใช้ข้อมูลตัวแปรทราฟฟิก (traffic parameter) ที่ต่างกัน และสนับสนุนการทำงานของโพรโทคอลสื่อสารข้อมูล (Data communication protocol) ที่ต่างกัน ซึ่งประเภทของระบบสื่อสารข้อมูลอาจแบ่งตามความสัมพันธ์ทางเวลาในการรับส่งข้อมูลได้เป็นแบบซิงโครนัส (Synchronous Transfer Mode, STM) [2] แบบอะซิงโครนัส (Asynchronous Transfer Mode, ATM) [3] และแบบกึ่งอะซิงโครนัส (Semi -Synchronous Transfer Mode) [4]

สำหรับการสื่อสารแบบซิงโครนัสนั้นสิ่งที่ต้องคำนึงถึงมากที่สุดคือ การควบคุมคาบเวลาในการรับส่งแพ็กเกตข้อมูลเพื่อให้แพ็กเกตเข้ามาตรงเวลาเพื่อให้แพ็กเกตได้รับการบริการในเวลาที่จองไว้ การหาเส้นทางที่รับประกันคุณภาพการบริการต้องพิจารณาเพื่อสามารถจองช่องสัญญาณเวลา (time-slot) ได้อย่างเหมาะสมกับการรับและส่งแพ็กเกตสำหรับแต่ละทราฟฟิก เช่น วิธีการหาเส้นทางใน การสื่อสารระบบไร้สาย [5]

ส่วนการสื่อสารแบบอะซิงโครนัส [6] นั้น การรับประกันคุณภาพการให้บริการจะขึ้นอยู่กับความต้องการของการประยุกต์ใช้งานในชั้นโปรแกรมประยุกต์ซึ่งมีการจัดแบ่งตามประเภทของการบริการ (service class) และเส้นทางที่ได้จะต้องเป็นเส้นทางที่สามารถสำรองทรัพยากร (resource reservation) ได้เหมาะสมกับความต้องการของประเภทการบริการ วิธีการหาเส้นทางสำหรับการสื่อสารแบบอะซิงโครนัสนั้นมีอยู่หลายวิธี เช่น การหาเส้นทางโดยใช้การสลับแพ็กเกตตามข้อมูลเลเบิลเพื่อสนับสนุนการสื่อสารข้อมูลของโพรโทคอลหลากหลายประเภท (Multi Protocol Label Switching, MPLS) [7] และการหาเส้นทางแบบสั้นที่สุด (Open Shortest Path First, OSPF) [8]

การรับประกันคุณภาพการให้บริการจำเป็นต้องทำการพิจารณาตัวแปรทราฟฟิกด้วย เช่น ค่าเวลาหน่วงสูงสุด (maximum Cell Transfer Delay, CTD) ช่วงการแกว่งของเวลาหน่วง (Cell Delay Variation, CDV) และอัตราการสูญหายของแพ็กเกตข้อมูล (Cell Loss Ratio, CLR) ซึ่งวิธีการคำนวณตัวแปรเหล่านี้จะขึ้นอยู่กับความต้องการของการบริการแต่ละประเภท

ในการสื่อสารแบบกึ่งอะซิงโครนัสนั้นจะให้บริการแก่ทราฟฟิกแบบซิงโครนัสควบคู่ไปกับแบบอะซิงโครนัส การรับประกันคุณภาพการ ให้บริการจะแยกตามความต้องการของการประยุกต์ที่ต่างกันไป ระบบสื่อสารแบบกึ่งอะซิงโครนัสนั้นทราฟฟิกแบบซิงโครนัสหรือทราฟฟิกที่มีอัตราเร็วคงที่จะได้รับการรับประกันคุณภาพการให้บริการสูงสุด ต้องไม่เกิดการสูญหายของข้อมูล (no cell lost) ไม่เกิดการแกว่งของเวลาหน่วง (no delay jitter) และจะมีการจำกัดขอบเขตของเวลาหน่วง (delay bounded) ตัวอย่างทราฟฟิกประเภทนี้ได้แก่ ทราฟฟิกที่ต้องการการให้บริการแบบ CBR (Constant Bit Rate) ของระบบสื่อสารแบบอะซิงโครนัส ส่วนการรับประกันคุณภาพของทราฟฟิกที่มีอัตราเร็วไม่คงที่จะขึ้นอยู่กับรูปแบบของการประยุกต์ ซึ่งเป็นแบบต่อเนื่อง (Real time) หรือไม่ต่อเนื่อง (Non real time)

ตัวอย่างของการพัฒนาเพื่อรองรับการสื่อสารแบบกึ่งอะซิงโครนัส เช่น โพรโทคอลสื่อสารแบบ Q-STM [4] และ W-TDM [9] ซึ่งทั้งสองวิธีนี้จะพิจารณาเพื่อให้บริการโดยใช้วิธีการจัดการเฟรม (Frame-base strategy) [12] ซึ่งเป็นการจัดสรรเวลาในการให้บริการโดยแบ่งช่วงเวลาของการให้บริการออกเป็นเฟรมและแบ่งเวลาในเฟรมเป็นช่องสัญญาณเวลา ซึ่งการให้บริการทราฟฟิกที่ต้องการการรับประกันคุณภาพจะใช้การจองช่องสัญญาณเวลาแบบตายตัวในเฟรมขนาดคงที่แล้วจึงใช้ข้อมูลการจองช่องสัญญาณในเฟรมนี้ไปพิจารณาเพื่อให้บริการในกระบวนการจัดลำดับแพ็กเกตเพื่อส่งออก (scheduling process) เมื่อทำการจองช่องสัญญาณเวลาเสร็จแล้วจะต้องเก็บข้อมูลของการจองไว้ในตารางข้อมูลการจัดลำดับแพ็กเกตเพื่อส่งออกหรือตารางวงจรเสมือน ขอบเขตของการพิจารณาเพื่อให้บริการในตารางวงจรเสมือนโดยทั่วไปนั้นจะเข้าถึงและตรวจสอบตารางวงจรเสมือนได้ทั้งตารางหรือไม่มีการจำกัดขอบเขตของการพิจารณา (Non-window base scheduling) ซึ่งวิธีการแบบนี้มีข้อดีคือ หากแพ็กเกตเข้ามาไม่ตรงกับตำแหน่งที่ได้จองเอาไว้ก็จะยังคงได้รับการบริการ แต่จะมีข้อเสียคือ ใช้เวลาในการพิจารณาในขั้นตอนนี้นานเกินไปและต้องใช้ความซับซ้อนในการจัดการเก็บแพ็กเกตลงบัฟเฟอร์เพื่อรอส่งออก

โพรโทคอลสื่อสารแบบ W-TDM นั้นจะพิจารณาในเรื่องการจัดการเฟรมและมีการกำหนดช่วงเวลาให้บริการ (service window) หรือเรียกว่าวินโดว์ รวมทั้งมีการสร้างบัฟเฟอร์เพื่อรองรับการมาไม่ตรงตำแหน่งช่องสัญญาณเวลา ซึ่งข้อดีของหลักการนี้คือเพิ่มความยืดหยุ่นของช่วงเวลายอมรับการให้บริการ ลดเวลาหน่วงในการส่งผ่านข้อมูลทั้งเส้นทาง ลดการแกว่งของเวลาหน่วง และลดเวลาหน่วงในกระบวนการเข้าถึงข้อมูลในตารางวงจรเสมือนในกระบวนการการจัดลำดับ

แพ็กเกตเพื่อส่งออกโดยใช้การจำกัดขอบเขตของวินโดว์ ในหลักการของ W-TDM นั้นหากแพ็กเกตเข้ามาตรงเวลาก็จะได้รับการบริการทันที หรือหากมาไม่ตรงตำแหน่งช่องสัญญาณเวลาที่จองไว้แต่ยังอยู่ในวินโดว์ก็จะยังได้รับการบริการ ดังนั้นการจองช่องสัญญาณเวลาในตำแหน่งที่เหมาะสมจึงเป็นเรื่องสำคัญมาก ดังนั้นหากสามารถระบุตำแหน่งที่ซิงโครนัสระหว่างเฟรมของโหนดในเส้นทางได้ก็จะสามารถรับประกันการรับส่งแพ็กเกตได้ตลอดเส้นทาง ลดอัตราการถูกปฏิเสธในขั้นตอนการสร้างเส้นทางเชื่อมโยงและขออนุมัติเส้นทางด้วย

วิธีการที่จะคำนวณหาตำแหน่งว่างที่ซิงโครนัสระหว่างเฟรมของโหนดในเส้นทางนั้นจะนำข้อมูลสถานะการใช้งานช่องสัญญาณเวลามาใช้ในกระบวนการคำนวณหาเส้นทางล่วงหน้า (Pre-computed routing) เพื่อให้เลือกเส้นทางได้เหมาะสมและลดโอกาสที่จะถูกปฏิเสธในกระบวนการสร้างเส้นทางเชื่อมโยงและขออนุมัติเส้นทางให้น้อยลง ข้อมูลสถานะการใช้งานช่องสัญญาณเวลานี้จะต้องแจ้งให้เราท์เตอร์ทราบเป็นระยะ ทั้งนี้เพื่อให้การคำนวณหาเส้นทางมีความแม่นยำมากขึ้น

ในงานวิจัยนี้ได้นำเสนอวิธีการหาเส้นทางโดยใช้ข้อมูลตำแหน่งช่องสัญญาณเวลา โดยนำข้อมูลสถานะการใช้งานช่องสัญญาณเวลามาพิจารณาร่วมกับตัวแปรทราฟฟิก ในรูปแบบฟังก์ชันของภาระการใช้งานลิงค์ (link cost) แล้วจึงทำการหาเส้นทางแบบสั้นที่สุด เมื่อได้เส้นทางแล้วจึงทำการเทียบบิตข้อมูลสถานะการใช้งานช่องสัญญาณเวลา (Slot Usage Bit Mapping, SUBM) เพื่อหาตำแหน่งช่องสัญญาณเวลาว่าง ในตำแหน่งที่ทำให้เกิดการซิงโครนัสของการรับส่งแพ็กเกตระหว่างแต่ละสวิตซิ่งโหนดในเส้นทาง สวิตซิ่งโหนดทุกตัวในระบบต้องส่งข้อมูลสถานะการใช้งานช่องสัญญาณเวลาไปยังเราท์เตอร์ทุกตัว เพื่อแจ้งข้อมูลสถานะการใช้งานช่องสัญญาณเวลาอย่างเป็นระยะเพื่อให้มีการปรับปรุงข้อมูลสถานะล่าสุดของระบบ ดังนั้นจึงเลือกพัฒนาอัลกอริทึมนี้เพิ่มลงในวิธีการหาเส้นทางแบบสั้นที่สุด (Shortest Path First) ซึ่งมีการส่งข้อมูลแจ้งสถานะการใช้งานเส้นทางเชื่อมโยง (Link state advertisement) อย่างเป็นระยะอยู่แล้ว ดังนั้นจึงสามารถแนบข้อมูลสถานะการใช้งานช่องสัญญาณ เวลาไปด้วยได้

อัลกอริทึมนี้ได้ถูกจำลองลงในระบบสื่อสารซึ่งมีการจองช่องสัญญาณเวลาตายตัวในเฟรมขนาดคงที่ด้วยวิธีการ W-TDM จากการทดลองพบว่าวิธีการหาเส้นทางที่มีการนำข้อมูลสถานะการใช้งานช่องสัญญาณเวลามาพิจารณาสามารถเลือกเส้นทางได้เหมาะสม มีอัตราการถูกปฏิเสธในกระบวนการการร้องขอเพื่อเชื่อมโยงเส้นทางน้อยกว่าวิธีการหาเส้นทางที่ไม่ได้ใช้ข้อมูลสถานะการใช้งานช่องสัญญาณเวลา

1.2 ความสำคัญของปัญหา

การรับประกันคุณภาพในระบบสื่อสารข้อมูลนั้นต้องพิจารณาตัวแปรทราฟฟิกที่มีผลต่อความต้องการในการเชื่อมโยงทราฟฟิก ดังนั้นการเลือกเส้นทางจะต้องพิจารณาตัวแปรทราฟฟิกที่

เกี่ยวข้องให้เป็นตัวแปรควบคุม แล้วใช้ตัวแปรหลักซึ่งมีผลต่อการรับประกันคุณภาพการให้บริการในการตัดสินใจขั้นสุดท้าย ทั้งนี้จะขึ้นอยู่กับความต้องการการรับประกันคุณภาพการให้บริการของแต่ละทราฟฟิก การพิจารณาตัวแปรทุกตัวนั้นจะทำเมื่อต้องการรองรับการให้บริการหลายประเภท เรียกวิธีการหาเส้นทางที่ต้องพิจารณาตัวแปรทราฟฟิกตามความต้องการในการรับประกันคุณภาพการบริการนี้เรียกว่า QoS Routing ซึ่งการพิจารณาเส้นทางที่เป็นไปได้จะทำโดยตัดเส้นทางที่ไม่ต้องการออกไปเหลือเพียงเส้นทางที่อยู่ในความควบคุมแล้วพิจารณาตัวแปรหลักเพื่อเลือกเส้นทางที่ต้องการ

การสื่อสารที่มีการกำหนดช่องสัญญาณเวลาแบบตายตัวในเฟรมขนาดคงที่นั้นการรู้ข้อมูลแบนด์วิดท์ที่เหลืออยู่เพียงอย่างเดียวนั้นไม่เพียงพอที่จะบอกได้ว่าตำแหน่งช่องสัญญาณเวลาที่จะถูกสำรองให้ในเส้นทางนั้นเหมาะสมกับการรับส่งข้อมูลและเหมาะสมกับความต้องการหรือไม่ เนื่องจากตำแหน่งช่องสัญญาณเวลาที่ว่างอยู่ของแต่ละสวิตช์ในโหนดนั้นอาจจะอยู่ในตำแหน่งที่ไม่ซิงโครนัสกัน ซึ่งจะมีผลต่อการถูกปฏิเสธในขั้นตอนการสร้างเส้นทางเชื่อมโยงและขออนุมัติเส้นทาง เป็นผลให้เวลาหน่วงในกระบวนการสร้างเส้นทางเชื่อมโยงเพิ่มมากขึ้น ดังนั้นการนำข้อมูลการใช้งานช่องสัญญาณเวลา (time-slot usage information) ของทุกโหนดในเส้นทางมาใช้ตั้งแต่ขั้นตอนการหาเส้นทางจะทำให้เลือกเส้นทางได้เหมาะสมมากขึ้น สามารคำนวณหาจำนวนแบนด์วิดท์ที่เพียงพอกับความต้องการได้โดยพิจารณาจากช่องสัญญาณเวลาว่าง และจะสำรองช่องสัญญาณเวลาในตำแหน่งที่เหมาะสมได้ด้วย วิธีการนี้สามารถนำไปใช้กับการสื่อสารที่มีการจองช่องสัญญาณเวลาตายตัวในเฟรมขนาดคงที่ได้ โดยเฉพาะในระบบสื่อสารที่ใช้โพรโทคอลสื่อสารแบบ W-TDM นั้นสามารถรับประกันการให้บริการแก่ทราฟฟิกที่ต้องการการรับประกันคุณภาพการให้บริการได้เป็นอย่างดี

1.3 วัตถุประสงค์ของวิทยานิพนธ์

เพื่อพัฒนาวิธีการหาเส้นทางที่เหมาะสมสำหรับระบบสื่อสารที่มีการกำหนดช่องสัญญาณเวลาตายตัวในเฟรมขนาดคงที่ โดยจะตัดสินใจเลือกเส้นทางที่เหมาะสมจากการพิจารณาตำแหน่งว่างที่ซิงโครนัสระหว่างโหนดในเส้นทาง ซึ่งพิจารณาได้จากการเทียบบิตข้อมูลสถานะการใช้งานช่องสัญญาณเวลาของทุกโหนดในเส้นทาง การเทียบบิตนั้นจะเป็นขั้นตอนสุดท้ายเพื่อตรวจสอบการรับประกันคุณภาพการให้บริการของเส้นทางที่ได้จากอัลกอริทึมการหาเส้นทางแบบสั้นที่สุด ซึ่งพิจารณาจากค่าภาระการใช้งานลิงค์ และฟังก์ชันของค่าภาระการใช้งานลิงค์นี้จะประกอบด้วยตัวแปรทราฟฟิกควบคุมที่เป็นเวลาหน่วงในการส่งผ่านสายสื่อสารและแบนด์วิดท์จากการคำนวณตำแหน่งว่างที่ซิงโครนัสระหว่าง 2 โหนดใด ๆ ที่เชื่อมโยงโดยสายสื่อสารดังกล่าว

1.4 ทฤษฎีและหลักการที่ใช้ในวิทยานิพนธ์

1.4.1 วิธีการหาเส้นทาง

จุดประสงค์หลักของการพัฒนาอัลกอริทึมการหาเส้นทางแบบนี้เพื่อรักษาการซิงโครนัสในเส้นทางเพื่อให้ตรงกับการรับประกันคุณภาพสำหรับการสื่อสารที่มีการกำหนดช่องสัญญาณเวลาตายตัวในเฟรมขนาดคงที่ โดยได้นำวิธีการอ้างถึงสถานะการใช้งานตำแหน่งช่องสัญญาณเวลาด้วยบิต 0 และบิต 1 ซึ่งถูกนำมาใช้กันอย่างแพร่หลายในการพิจารณาเพื่อจองช่องสัญญาณเวลาในการสื่อสารแบบไร้สายและแบบดาวเทียมเพื่อให้มั่นใจว่ามีตำแหน่งที่เหมาะสมกับความต้องการ การนำบิตข้อมูลตำแหน่งช่องสัญญาณเวลาในเฟรมมาพิจารณานี้ก็จะต่างกันไปตามอัลกอริทึมการหาเส้นทางและการจองช่องสัญญาณเวลาเพื่อให้เหมาะกับการสื่อสารแต่ละประเภท ซึ่งอาจจะแตกต่างกันที่วิธีการแจ้งข้อมูล วิธีการนำข้อมูลมาคำนวณโดยจะต่างกันตามจุดประสงค์ของการสื่อสารแต่ละประเภท ด้วยเหตุนี้จึงมีอัลกอริทึมที่ใช้ข้อมูลสถานะการใช้งานช่องสัญญาณเวลามาพิจารณาอย่างหลากหลาย เช่น การสื่อสารไร้สายแบบมีหลายจุดเชื่อมโยง (Multihop mobile wireless network) [15]

ในงานวิจัยนี้ได้พัฒนาอัลกอริทึมการหาเส้นทางโดยเลือกวิธีการหาเส้นทางแบบสั้นที่สุดมาปรับปรุงเพื่อเลือกเส้นทางที่มีความเป็นไปได้จำนวนหนึ่งจากการพิจารณาค่าภาระการใช้งานเส้นทางเชื่อมโยง ส่วนที่จะทำการปรับปรุงเพิ่มคือการคำนวณค่าภาระการใช้งานลิงค์ซึ่งจะเป็นการนำตัวแปรทราฟฟิกควบคุมมาคำนวณล่วงหน้า (pre–calculate routing) ดังในตัวอย่าง [17] ซึ่งได้นำตัวแปรทราฟฟิกที่สำคัญมากำหนดขอบเขตของค่าต่ำสุดหรือสูงสุดที่ยอมรับได้ และใช้ขอบเขตนี้ในการเลือกเส้นทางที่เหมาะสม แต่งานวิจัยนี้ได้นำตัวแปรทราฟิกควบคุมมาพิจารณาในรูปแบบฟังก์ชันของภาระการใช้งานลิงค์และเพื่อลดความยุ่งยากในการควบคุมตัวแปรอื่น ๆ จึงสนใจเพียงตัวแปรแบนด์วิดท์และเวลาหน่วง แล้วเพิ่มการพิจารณาแบนด์วิดท์ด้วยการนับเฉพาะจำนวนช่องสัญญาณเวลาว่างที่ซิงโครนัสกันเท่านั้น ซึ่งจะการพิจารณาช่องสัญญาณเวลาว่างที่ซิงโครนัสตามข้อมูลสถานะการใช้งานช่องสัญญาณเวลาที่ถูกส่งมาเป็นระยะ หากใช้จำนวนช่องสัญญาณเวลาว่างทั้งหมดมาพิจารณาอาจจะได้เส้นทางที่ไม่เหมาะสมและอาจเกิดการปฏิเสธในกระบวนการร้องเพื่อเชื่อมโยงเส้นทางซึ่งจะทำให้เพิ่มเวลาหน่วงในกระบวนการร้องขอเชื่อมโยงเส้นทางได้ และในการเลือกเส้นทางที่สั้นที่สุดนั้นในแต่ละรอบของการพิจารณาเส้นทางที่สั้นที่สุดจากโหนดต้นทางไปยังโหนดปลายทางใด ๆ นั้นจะเก็บค่าภาระการใช้งานลิงค์ระหว่างโหนดไว้เพื่อนำมาคำนวณเป็นค่าภาระการใช้งานลิงค์ของทั้งเส้นทาง แล้วจึงเก็บเส้นทางที่เป็นไปได้เรียงลำดับตามค่าภาระการใช้งานลิงค์จากน้อยไปมาก จากวิธีการนี้จะได้เส้นทางที่เป็นไปได้จากโหนดต้นทางไปยังปลายทางจำนวนหนึ่ง แล้วจึงนำเส้นทางที่เป็นไปได้เหล่านี้มาพิจารณาเทียบบิตข้อมูลสถานะการใช้งานช่อง

สัญญาณเวลาอีกครั้งเพื่อเลือกเส้นทางที่เหมาะสม ซึ่งกระบวนการพิจารณาเพื่อเทียบบิตข้อมูลจะทำให้ได้ตำแหน่งช่องสัญญาณเวลาที่สามารถจองได้ในทุกโหนด

การจองช่องสัญญาณเวลานั้นจะสัมพันธ์กับการจัดลำดับแพ็กเกตเพื่อส่งออก ดังนั้นการเทียบบิตเพื่อให้ได้เส้นทางที่เหมาะสมและได้ตำแหน่งช่องสัญญาณเวลาที่สามารถใช้งานได้นี้จึงต้องสอดคล้องกับวิธีการของการจองช่องสัญญาณเวลาและการจัดลำดับแพ็กเกตเพื่อส่งออกด้วย ซึ่งจะอธิบายโดยละเอียดในบทที่ 3 ชุดข้อมูลบิตสถานะการใช้งานช่องสัญญาณเวลานี้จะมีจำนวนบิตเท่ากับจำนวนของช่องสัญญาณเวลาในเฟรมและใช้บิต 1 แทนสถานะช่องสัญญาณเวลาว่างและใช้บิต 0 แทนสถานะช่องสัญญาณเวลาที่กำลังให้บริการ การเทียบบิตจะใช้การเลื่อนตำแหน่งช่องสัญญาณเวลาให้ตรงกับตำแหน่งที่ซิงโครนัสกันก่อน จากนั้นจะใช้การ AND บิตข้อมูลของทั้งเฟรม หากผลการAND ทั้งเฟรมเป็น 0 แล้วถือว่าไม่มีช่องสัญญาณเวลาว่างที่ซิงโครนัส หากผลการ AND มีตำแหน่งที่เป็น 1 แสดงว่าตำแหน่งดังกล่าวเป็นตำแหน่งที่ซิงโครนัสกันจะถูกนำมาพิจารณาตามแบนด์วิดท์ที่ร้องขอว่าต้องการใช้งานกี่ช่องสัญญาณเวลา หากจำนวนช่องสัญญาณเวลาว่างเพียงพอต่อความต้องการ เส้นทางดังกล่าวก็จะเป็นเส้นทางที่เหมาะสม

เนื่องจากเส้นทางที่น่าจะเป็นไปได้มีอยู่หลายเส้นทางดังนั้นการตัดสินใจเลือกเส้นทางมาเพื่อพิจารณาจะตัดสินใจเลือกเส้นทางที่มีค่าเวลาหน่วงน้อยที่สุดและมีจำนวน Hop น้อยที่สุดเป็นอันดับแรก เนื่องจากเส้นทางบางเส้นทางอาจจะมีช่วงของเส้นทางที่ซ้ำกัน ดังนั้นหากต้องพิจารณาหลายเส้นทางก็สามารถใช้ข้อมูลการเทียบบิตเดิมมาใช้ได้ หรือจะเปรียบเทียบเส้นทางที่เหมาะสมกว่าก็สามารถพิจารณาเพียงช่วงของเส้นทางที่ไม่ซ้ำแนวเดียวกันได้ ซึ่งจะทำให้ใช้เวลาน้อยลงในการพิจารณา การหาเส้นทางสำหรับการสื่อสารแบบที่การกำหนดช่องสัญญาณเวลาตายตัวในเฟรมขนาดคงที่นั้นการซิงโครนัสสำคัญที่สุด ซึ่งเส้นทางที่ได้จากวิธีการหาเส้นทางแบบนี้นั้นจะรับประกันการซิงโครนัสได้และยังมีแบนด์วิดท์เพียงพอในขณะที่มีเวลาหน่วงน้อยและมีจำนวนจุดเชื่อมโยงน้อยด้วย และในงานวิจัยนี้ได้พัฒนาวิธีการหาเส้นทางลงในระบบการสื่อสารที่ใช้การจัดลำดับแพ็กเกตเพื่อส่งออกแบบ W-TDM ซึ่งเป็นการสื่อสารแบบกึ่งอะซิงโครนัสที่สนับสนุนทราฟฟิกที่มีการจองช่องสัญญาณเวลาแบบตายตัวในเฟรมขนาดคงที่ ซึ่งหากแพ็กเกตเข้าในช่วงเวลายอมรับให้บริการแพ็กเกตจะได้รับการรับประกันในการบริการอย่างแน่นอน

1.4.2 **หลักการ** W-TDM

วิธีการจัดลำดับแพ็กเกตเพื่อส่งออกแบบ W-TDM นั้นมีการกำหนดขอบเขตของวินโดว์บนเฟรม การทำงานของวินโดว์จะเป็นการเพิ่มความยืดหยุ่นของช่วงเวลาการพิจารณาเพื่อให้บริการและสามารถจำกัดขอบเขตการพิจารณาได้จากการกำหนดขนาดของวินโดว์ การทำงานของวินโดว์นั้นจะเลื่อนไปทีละช่องสัญญาณเวลาในทิศทางเดิมเสมอและมีการหมุนวนในช่วงเฟรม ซึ่งวิธีการ

หมุนวนนี้จะคล้ายกับการจัดการแบบ Round Robin [15] แพ็กเกตที่เข้ามาในช่วงเวลาของวินโดว์จะได้รับการบริการโดยส่งข้อมูลผ่านสวิตช์ไปยังบัฟเฟอร์ขาออก และรอส่งออกต่อไป หากแพ็กเกตเข้ามาไม่ตรงตำแหน่งช่องสัญญาณเวลาที่จองไว้ และตำแหน่งให้บริการไม่ได้อยู่ในช่วงของวินโดว์แล้วแพ็กเกตนั้นจะถูกลบทิ้ง ซึ่งทำให้เกิดการสูญหายของแพ็กเกตแต่ทั้งนี้จะขึ้นอยู่กับลักษณะการกระจายและพฤติกรรมของทราฟฟิกนั้น ดังนั้นหากสามารถทำให้การรับส่งแพ็กเกตระหว่างสวิตช์ซิงโหนดเป็นแบบซิงโครนัส ก็จะลดการสูญหายของแพ็กเกตได้ ข้อดีของวิธีการ W-TDM คือ ช่วยเพิ่มความยืดหยุ่นให้กับการบริการแบบ TDM และจะใช้ช่องสัญญาณเวลาได้เต็มที่ในแกนเวลา วิธีการของ W-TDM นั้นทำให้เกิดความยืดหยุ่นในการให้บริการได้มากกว่าการบริการแบบเดิมของ TDM การกำหนดช่วงเวลาของวินโดว์จะทำให้การจัดการเป็นไปได้ง่ายยิ่งขึ้น และจะลดความยุ่งยากในการออกแบบการทำงานในสวิตช์ นอกจากนี้การให้บริการระบบ CBR ในหลักการของ W-TDM นั้นจะมีประสิทธิภาพมากขึ้น ทำให้ทราฟฟิกมีระบบระเบียบมากขึ้น คล้ายกับการให้บริการของ STM ในวิธีการ Q-STM และหลักการของวินโดว์ยังทำให้เกิดการรับประกันการบริการในการสื่อสารแบบ VBR และ ABR ด้วย โดยจำกัดขอบเขตของเวลาหน่วง (delay-bound) และลดการแกว่งของเวลาหน่วง ส่วนความน่าจะเป็นของการสูญหายแพ็กเกตจะลดลงได้โดยการใช้บัฟเฟอร์ร่วม งานวิจัยนี้สนใจเพียงทราฟฟิกที่ต้องการการรับประกันคุณภาพการบริการ ซึ่งต้องการจองช่องสัญญาณเวลาแบบตายตัวในเฟรมขนาดคงที่ จึงไม่ได้พิจารณาในส่วนของทราฟฟิกที่มีอัตราเร็วแบบไม่คงที่อื่น ๆ รายละเอียดของหลักการทำงานจะกล่าวถึงต่อไปในบทที่ 2

ข้อดีของหลักการ W-TDM มีดังนี้

1. ทุกทราฟฟิกจะได้รับการสำรองช่องสัญญาณเวลาพร้อมกับการรับประกันการบริการ

2. ทราฟฟิกที่ต้องการการรับประกันคุณภาพการให้บริการในเวลาที่กำหนดนี้ จะได้รับการบริการในลักษณะกึ่งอะซิงโครนัส ซึ่งจะทำให้ได้รับบริการและส่งออกไปยังโหนดถัดไป ณ เวลาที่มีสำรองไว้ในสวิตช์ซิ่งโหนด

3. หลักการของวินโดว์จะเพิ่มความยืดหยุ่นในการให้กับการบริการแบบ TDM เช่น กรณีที่มีแพ็กเกตเข้ามาถึงก่อนเวลาให้บริการ แต่อยู่ภายในขอบเขตของวินโดว์ก็จะยังได้รับการบริการไม่มีการสูญหายของแพ็กเกตข้อมูล

4. การสูญหายของแพ็กเกตจะถูกจัดการโดยการแยกการรับประกันการบริการ ตามความต้องการของทราฟฟิกแต่ละทราฟฟิก

5. หลักการทำงานของ W-TDM สามารถนำไปพัฒนาลงในการสื่อสารได้ทุกประเภท

1.5 ขอบเขตของวิทยานิพนธ์

1. นำเสนออัลกอริทึมการหาเส้นทางที่เหมาะสมสำหรับระบบการสื่อสารซึ่งมีการกำหนดช่องสัญญาณเวลาตายตัวในเฟรมขนาดคงที่ โดยนำข้อมูลสถานะการใช้งานช่องสัญญาณเวลามาพิจารณาร่วมกับตัวแปรทราฟฟิกควบคุมอื่น ๆ ในรูปแบบฟังก์ชันของภาระการใช้งานลิงค์ แล้วใช้ค่าภาระการใช้งานลิงค์ในการเลือกเส้นทางที่เป็นไปได้โดยใช้หลักการหาเส้นทางแบบสั้นที่สุด แล้วจึงเลือกเส้นทางที่เหมาะสมโดยการเทียบบิตข้อมูลสถานะการใช้งานช่องสัญญาณเวลาเพื่อเลือกเส้นทางที่เหมาะสมและเลือกตำแหน่งช่องสัญญาณเวลาที่ทำให้เกิดการซิงโครนัสในการรับส่งระหว่างเฟรมของแต่ละสวิตช์ในโหนดในเส้นทาง และกำหนดให้ทุกทราฟฟิกทำการร้องขอเพื่อสำรองเส้นทางโดยต้องการใช้งานแบนด์วิดท์ 1 ช่องสัญญาณเวลาใน 1 เฟรม

2. ออกแบบและพัฒนาอัลกอริทึมในข้อ 1 ลงในอุปกรณ์เครือข่ายจำลอง ได้แก่ อุปกรณ์สื่อสารปลายทาง เราท์เตอร์ และสวิตช์ และใช้โพรโตคอลสื่อสาร W-TDM ในกระบวนการจัดลำดับแพ็กเกตเพื่อส่งออก ซึ่งได้ถูกพัฒนาไว้ในอุปกรณ์สวิตช์และเราท์เตอร์จำลอง ทั้งนี้เพื่อเป็นการจำลองระบบสื่อสารที่มีการจองช่องสัญญาณเวลาแบบตายตัวในเฟรมขนาดคงที่ และติดตั้งอุปกรณ์เครือข่ายจำลองนี้เป็นระบบเครือข่ายลงในโปรแกรมจำลองการทำงานระบบเครือข่าย Opnet 6.0 แล้วเปรียบเทียบประสิทธิภาพของการหาเส้นทางระหว่างกรณีที่ใช้อัลกอริทึม SUBM กับกรณีที่ไม่ได้ใช้อัลกอริทึม SUBM

1.6 ขั้นตอนการทำวิทยานิพนธ์

1. กำหนดจุดประสงค์ หัวข้อ และขอบเขตการทำวิทยานิพนธ์

2. ศึกษาทฤษฎี หลักการพื้นฐาน ข้อมูลที่เกี่ยวข้อง แนวทางในการพัฒนาอัลกอริทึมและตั้งสมมติฐานเพื่อศึกษาอัลกอริทึมที่นำเสนอ

3. ออกแบบการทำงานของอุปกรณ์ในระบบเครือข่ายจำลองและสร้างมอดูลจำลองของอุปกรณ์เหล่านั้น แล้วติดตั้งอุปกรณ์เครือข่ายทั้งหมดเป็นระบบเครือข่ายจำลองบนโปรแกรมจำลองการทำงานระบบเครือข่าย Opnet

4. พัฒนาวิธีการหาเส้นทางแบบสั้นที่สุดลงในอุปกรณ์เราท์เตอร์จำลองแล้วเพิ่มอัลกอริทึม SUBM ลงในวิธีการหาเส้นทางแบบสั้นที่สุด แล้วจึงทำการจำลองเพื่อเปรียบเทียบประสิทธิภาพระหว่างวิธีการหาเส้นทางแบบสั้นที่สุดที่ใช้อัลกอริทึม SUBM กับกรณีที่ไม่ใช้อัลกอริทึม SUBM

5. สรุป วิเคราะห์ผลการทดลอง และนำเสนอแนวทางในการทำวิจัยต่อไป

6. จัดทำเอกสารประกอบวิทยานิพนธ์

1.7 โครงสร้างของวิทยานิพนธ์

เนื้อหาในวิทยานิพนธ์ฉบับนี้ประกอบด้วย

บทที่ 1 กล่าวถึงความเป็นมา ความสำคัญของปัญหา วัตถุประสงค์ ทฤษฎีและหลักการโดยย่อที่นำมาใช้ในงานวิจัย ขอบเขตของงานวิจัย ขั้นตอนการทำงาน และโครงสร้างของวิทยานิพนธ์

บทที่ 2 อธิบายถึงทฤษฎีและหลักการต่างที่นำมาพิจารณาประกอบในงานวิจัย

บทที่ 3 การออกแบบอุปกรณ์เครือข่ายและการพัฒนาอัลกอริทึม

บทที่ 4 การออกแบบระบบเครือข่ายจำลอง การจำลองและผลการจำลอง

บทที่ 5 สรุปวิเคราะห์ผลการจำลองและนำเสนอแนวทางในการวิจัยต่อไป

บทที่ 2
ทฤษฎีที่เกี่ยวข้องในงานวิจัย

2.1 อัลกอริทึมการค้นหาเส้นทาง (Routing Algorithm)

ปัจจุบันระบบสื่อสารข้อมูลมีการพัฒนาโปรแกรมประยุกต์เพื่อใช้งานระบบเครือข่ายมากมาย เช่น การประชุมระยะไกล (Teleconference) โทรศัพท์ผ่านทางอินเตอร์เน็ต (Internet phone) การเล่นเกมส์ผ่านทางเว็บไซต์ เป็นต้น การดำเนินกิจกรรมต่าง ๆ ผ่านระบบเครือข่ายเป็นการส่งผ่านข้อมูลจากที่หนึ่งไปยังอีกที่หนึ่งซึ่งอาจจะเป็นการส่งผ่านข้อมูลภายในระบบเครือข่ายเดียวกันหรือข้ามเครือข่ายไปยังระบบเครือข่ายที่อยู่ห่างกัน การส่งผ่านข้อมูลหลาย ๆ ประเภทในระบบเครือข่ายนั้นอุปกรณ์เราท์เตอร์ถือเป็นหัวใจสำคัญในการเชื่อมโยงเครือข่ายและทำกระบวนการส่งผ่านข้อมูลข้ามเครือข่ายไปยังเครื่องที่อยู่ปลายทางได้อย่างถูกต้องและมีประสิทธิภาพเนื่องจากระบบเครือข่ายในโลกนี้มีอยู่มากมายหลายระบบและหลายขนาดทั้งระบบเล็กและระบบใหญ่เชื่อมโยงกันอยู่มากมาย ดังนั้นการเดินทางของ ข้อมูลข้ามเครือข่ายไปยังปลายทางนั้น ถ้าไม่มีการกำหนดเงื่อนไขของการสื่อสารหรือกำหนดเส้นทางที่มีประสิทธิภาพแล้วจะทำให้เกิดการติดขัดในการส่งข้อมูลผ่านเครือข่าย ซึ่งอาจทำให้การสื่อสารข้อมูลเป็นไปอย่างล่าช้าหรือไปไม่ถึงปลายทางได้ การเชื่อมโยงระบบเครือข่ายที่อยู่ห่างไกลกันเข้าด้วยกันไม่ว่าเครือข่ายนั้นจะต่างหรือเหมือนกันทางด้านกายภาพก็ตาม จะมีการควบคุมในโปรโตคอลสื่อสารชั้นควบคุมเครือข่าย (Network layer) ซึ่งมีหน้าที่ควบคุมดูแลจัดส่งข้อมูลจากผู้ส่งถึงผู้รับปลายทางที่กำหนด การส่งข้อมูลจะทำได้โดยไม่ต้องสนใจว่าทางด้านกายภาพของระบบเครือข่ายที่เชื่อมต่ออยู่จะเป็นอย่างไร ซึ่งการทำงานของเราท์เตอร์นั้นมีการรับและส่งข้อมูลสถานะของระบบและค่าตัวแปรทราฟฟิกระหว่างกันเพื่อแลกเปลี่ยนข้อมูลเกี่ยวกับเส้นทางเป็นระยะตามเวลาที่กำหนดและเก็บข้อมูลนี้ไว้ใช้ในกระบวนการค้นหาเส้นทาง และเมื่อหาเส้นทางได้แล้วจะเก็บข้อมูลเส้นทางและข้อมูลทราฟฟิกที่เกี่ยวข้องกับเส้นทางโดยตรงไว้ในตารางเส้นทาง แต่เนื่องจากสถานะการทำงานภายในระบบเครือข่ายมีการเปลี่ยนแปลงตลอดเวลา ดังนั้นเส้นทางการส่งข้อมูลที่ดีที่สุดที่เลือกไว้อาจเป็นเส้นทางที่ใช้ไม่ได้ทั้งนี้ขึ้นอยู่กับการปรับค่าข้อมูลสถานะการทำงานของระบบล่าสุดให้แก่อุปกรณ์เครือข่ายในระบบนั่นเอง ซึ่งข้อมูลและวิธีการแลกเปลี่ยนข้อมูลสถานะการทำงานในระบบเครือข่ายนี้จะต่างกันไปตามวิธีการหาเส้นทาง และวิธีการค้นหาเส้นทางแต่ละแบบก็จะมีความเหมาะสมกับระบบสื่อสารที่แตกต่างกันไป ทั้งนี้ต้องสนับสนุนความต้องการของการประยุกต์ใช้งานในระบบสื่อสารข้อมูล ดังนั้นการเลือกวิธีการค้นหาเส้นทางจะต้องเหมาะสมกับความต้องการของการประยุกต์ใช้งาน

หลักการสำคัญของอัลกอริทึมการหาเส้นทางทุกแบบ

1. ต้องได้เส้นทางที่สามารถส่งข้อมูลถึงปลายทางได้อย่างถูกต้อง เนื่องจากระบบโครงข่ายจริงที่เชื่อมโยงกันอยู่นั้นประกอบด้วยเครือข่ายน้อยใหญ่และเครือข่ายย่อยมากมายดังนั้นต้องแน่ใจว่าเส้นทางที่เลือกเป็นเส้นทางที่สามารถส่งข้อมูลไปยังปลายทางได้อย่างถูกต้อง

2. กระบวนการหาเส้นทางต้องมีความสะดวก รวดเร็ว ใช้ทรัพยากรน้อย การหาเส้นทางนั้นหากได้เส้นทางที่ไม่ดีอาจทำให้เกิดการปฏิเสธในกระบวนการร้องขอเพื่อเชื่อมโยงเส้นทางได้แล้วต้องมีการหาเส้นทางอื่นใหม่ซึ่งจะทำให้เกิดความล่าช้าในการเชื่อมโยงเส้นทางสื่อสารได้ วิธีการคำนวณก็เป็นอีกสาเหตุหนึ่งที่ทำให้เสียเวลาในการพิจารณา หากมีความยุ่งยากและซับซ้อนมากก็อาจจะต้องใช้เวลาในการประมวลผลนานเช่นกัน ในกระบวนการหาเส้นทางนั้นอาจจะมีการส่งแพ็กเกตออกไปในระบบซึ่งหากมากเกินไปก็จะเป็นการเพิ่มทราฟฟิกในระบบได้ การใช้งานทรัพยากรของส่วนต่าง ๆ ในระบบ เช่น การใช้งานแบนด์วิดท์ การใช้งานหน่วยความจำและหน่วยประมวลผลในเราท์เตอร์โหนดเป็นต้น ซึ่งวิธีการหาเส้นทางที่ดีนั้นควรจะใช้เวลาและทรัพยากรในกระบวนการหาเส้นทางให้น้อยในขณะที่ได้เส้นทางที่มีประสิทธิภาพและต้องสนับสนุนการทำงานเป็นลำดับชั้น (Hierarchical) นั่นคือสามารถรองรับการหาเส้นทางสำหรับระบบเครือข่ายทุกลำดับชั้นได้อย่างถูกต้อง

3. รองรับการแก้ไขข้อผิดพลาดหรือปัญหาที่จะเกิดขึ้นในเส้นทางได้ ตัวอย่างของข้อผิดพลาดหรือปัญหาที่เกิดขึ้น เช่น กรณีที่เส้นทางเกิดความเสียหาย หรือกรณีที่สวิตช์หรือเราท์เตอร์เกิดความเสียหายจะต้องมีวิธีการที่แก้ปัญหาซึ่งอาจจะเป็นการกู้เส้นทาง (Route recovery) หรือเป็นการหาเส้นทางใหม่ที่รวดเร็วหรือช่วยให้การส่งแพ็กเกตไม่ถูกตัดขาดได้

4. เส้นทางที่ได้ควรมีเสถียรภาพในการรองรับภาระโหลด ซึ่งโหลดในที่นี้คือปริมาณทราฟฟิกและความต้องการใช้งานของทราฟฟิกซึ่งบางครั้งหากเป็นทราฟฟิกที่มีอัตราเร็วไม่คงที่อาจจะมีการส่ง ทราฟฟิกออกมาในระบบมากหรือน้อย ดังนั้นเส้นทางที่ได้ต้องรองรับอัตราเร็วสูงสุดและรองรับปริมาณ ทราฟฟิกที่จะเข้ามาร่วมใช้งานเส้นทางนั้นได้

5. มีความเป็นธรรมสำหรับผู้ขอใช้บริการหรือมีความเป็นธรรมในการให้บริการแก่ทุกทราฟฟิกที่เข้ามาใช้งาน ซึ่งเงื่อนไขในการร้องขอเพื่อใช้งานเส้นทางของแต่ละทราฟฟิกนั้นจะแตกต่างกันไป ดังนั้นในกระบวนการหาเส้นทางควรเป็นการเตรียมเส้นทางที่สามารถรองรับและให้ความเป็นธรรมแก่การบริการทราฟฟิกต่าง ๆ ได้ และนอกจากนี้ยังต้องสนับสนุนการรับประกันคุณภาพการให้บริการแก่บริการหลาย ๆ ประเภทในขณะเดียวกันได้ด้วย (QoS support for Integrated Service)

ข้อมูลที่จะนำมาพิจารณาในการหาเส้นทาง (Routing Metrics)

การเลือกเส้นทางที่ดีที่สุดนั้นต้องพิจารณาข้อมูลเส้นทางด้วยซึ่งข้อมูลสำคัญที่นำมาพิจารณาจะประกอบด้วย

1. Path length ซึ่งเป็นข้อมูลที่ใช้กันโดยทั่วไป การหาเส้นทางบางวิธีนั้นจะอนุญาตให้ผู้ดูแลระบบเป็นผู้กำหนดให้แต่ละเส้นทางในรูปแบบของค่าภาระการใช้งานลิงค์ ซึ่งในที่นี้จะเป็นค่าภาระการใช้งานลิงค์รวมของทั้งเส้นทาง อัลกอริทึมการหาเส้นทางบางอย่างอาจจะใช้การนับจำนวนจุดเชื่อมโยง (hop count) ในการพิจารณาเส้นทางเป็นต้น

2. ความน่าเชื่อถือ (Reliability) เส้นทางบางเส้นทางอาจมีปัญหาหรือมีความเสียหายบ่อยกว่าเส้นทางอื่น ซึ่งหลังจากที่เกิดความเสียหายระบบต้องสามารถกู้เส้นทางได้อย่างง่ายดายและรวดเร็ว ดังนั้นตัวประกอบ (factor) ที่เกี่ยวข้องจะต้องแจ้งให้อุปกรณ์เครือข่ายทราบด้วย

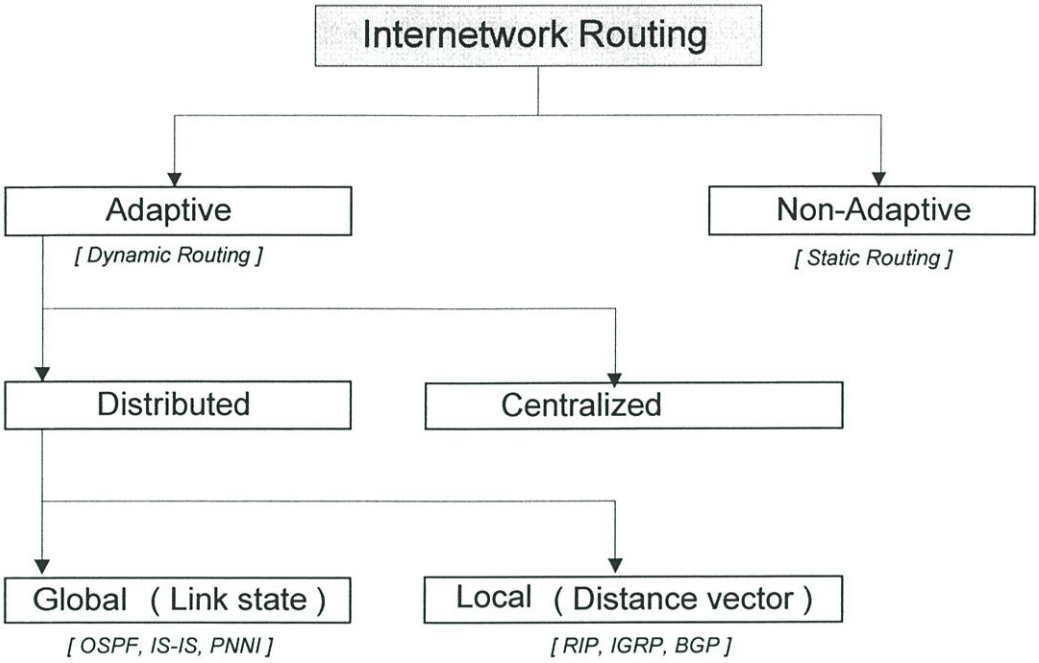
3. เวลาหน่วงในกระบวนการหาเส้นทาง (Routing delay) ซึ่งจะอ้างถึงเวลาที่ต้องใช้จนกว่าจะสามารถส่งแพ็กเกตข้อมูลจากต้นทางไปยังปลายทางได้ เวลาหน่วงนี้จะขึ้นอยู่กับตัวประกอบหลายตัว เช่น แบนด์วิดท์ของเส้นทางเชื่อมโยงระหว่างโครงข่าย การเรียงลำดับและรอคอยในแถวคอยในแต่ละทางเข้าออก (port) ของเราท์เตอร์ต่าง ๆ ตลอดทั้งเส้นทาง ความคับคั่งในเส้นทาง รวมทั้งอาจเกิดจากปัญหาพื้นฐานของสายสื่อสารเอง ซึ่งต้องพิจารณาตัวประกอบทั้งหมดที่เกี่ยวข้องจึงจะหาเวลาหน่วงที่เกิดขึ้นได้

4. การอ้างถึงแบนด์วิดท์ (Bandwidth) เป็นข้อมูลที่จะบอกถึงความจุของลิงค์ เช่น อีเทอร์เนตลิงค์ (Ethernet link) 10 เมกะบิตต่อวินาทีย่อมดีกว่าสายเช่า (leased line) 64 กิโลบิตต่อวินาที แต่ก็ไม่ใช่ว่าเส้นทางที่มีเร็วกว่าจะดีกว่าเสมอไป เพราะเส้นทางที่เร็วกว่าแต่มีทราฟฟิกเยอะมากก็อาจจะช้ากว่าอีกเส้นทางที่มีความเร็วน้อยกว่าได้ ดังนั้นเส้นทางที่ดีกว่าจะต้องพิจารณาอย่างอื่นด้วย

5. ภาระ (Load) ในที่นี้มีความหมายหลายอย่าง เช่น การใช้งานหน่วยประมวลผล (CPU utilization) ความเร็วในกระบวนการจัดการแพ็กเกต ซึ่งตัวเราท์เตอร์เองจะต้องมีความสามารถในการตรวจสอบทรัพยากรของตัวเอง

6. ค่าใช้จ่ายในการสื่อสาร (Communication cost) ส่วนนี้ก็สำคัญเนื่องจากผู้ใช้บางรายสนใจเรื่องค่าใช้จ่ายมากกว่าประสิทธิภาพของเส้นทาง แม้ว่าเส้นทางที่เลือกอาจจะมีเวลาหน่วงมากกว่าแต่ก็ประหยัดค่าใช้จ่ายได้มากกว่าการจ่ายเพื่อเชื่อมโยงเส้นทางตามเวลาใช้งาน

การเลือกเส้นทางที่ดีที่สุดนั้นต้องพิจารณาข้อมูลและความต้องการหลาย ๆ อย่างทั้งความต้องการของผู้ใช้เอง ความต้องการในการใช้งานของระบบ ความต้องการเพื่อให้สนับสนุนการทำงานของงานประยุกต์บางอย่าง เป็นต้น ซึ่งจะได้กล่าวถึงวิธีการหาเส้นทางแบบต่าง ๆ ต่อไป



รูปที่ 2.1 อัลกอริทึมการหาเส้นทาง

อัลกอริทึมการค้นหาเส้นทางสามารถแบ่งหมวดหมู่ตามลักษณะการทำงานได้ดังรูปที่ 2.1 ซึ่งวิธีการหาเส้นทางแต่ละแบบต่างก็มีกระบวนการทำงานที่ต่างกันซึ่งสามารถอธิบายการทำงานของวิธีการต่าง ๆ ได้ดังนี้

2.1.1. **การหาเส้นทางแบบไม่มีการปรับตัว** (Non-Adaptive Routing)

การติดตั้งเส้นทางจะทำโดยผู้ดูแลระบบมีการกำหนดเส้นทางที่แน่นอนตายตัว โดยผู้ดูแลระบบเป็นผู้พิจารณาเส้นทางที่เหมาะสม แล้วป้อนข้อมูลลงในตารางเส้นทาง ซึ่งจะมีค่าตายตัวไปตลอดจนกว่าผู้ดูแลระบบจะทำการเปลี่ยนแปลงใหม่ การหาเส้นทางแบบนี้จะเหมาะสำหรับระบบเครือข่ายที่มีขนาดเล็กไม่ซับซ้อน เช่น การหาเส้นทางแบบสถิตย์ (Static Routing) [16] เป็นต้น

ข้อดีของวิธีการนี้คือ

1. สะดวกในการใช้งาน สำหรับเครือข่ายขนาดเล็ก
2. ไม่ต้องใช้การคำนวณหาเส้นทางทำให้ลดภาระในการประมวลผล
3. ไม่ต้องส่งข้อมูลระหว่างเราท์เตอร์ข้างเคียง ทำให้ระบบประหยัดแบนด์วิดท์มากขึ้น

แต่มีข้อเสียในการใช้งาน เช่น

1. ไม่เหมาะสำหรับเครือข่ายขนาดใหญ่ ถ้ามีเราท์เตอร์ในระบบมากเกินไปผู้ดูแลระบบจะต้องใช้ความสามารถในการกำหนดตารางการหาเส้นทางขนาดใหญ่และซับซ้อนซึ่งอาจทำให้เกิดความผิด พลาดหรือกำหนดเส้นทางที่ไม่เหมาะสม

2. หากมีการเปลี่ยนแปลงโครงสร้างภายในเครือข่าย ผู้ดูแลระบบจะต้องคำนวณและป้อนข้อมูลของตารางการหาเส้นทางใหม่ทั้งหมด

3. หากเราท์เตอร์หรือเส้นทางใดถูกตัดขาดอาจส่งผลถึงการหาเส้นทางภายในเครือข่ายทั้งหมดเพราะเราท์เตอร์ไม่สามารถหาเส้นทางใหม่เองได้

4. ไม่สามารถบริหารการกระจายโหลดไปยังส่วนต่าง ๆ ได้อย่างมีประสิทธิภาพ

2.1.2. **การหาเส้นทางแบบมีการปรับตัว** (Adaptive Routing)

การหาเส้นทางวิธีนี้บางครั้งเรียก Dynamic Routing ซึ่งจะอาศัยการศึกษาข้อมูลการสื่อสารของเราท์เตอร์หรือสวิตชิ่งโหนดอื่น ๆ ในระบบ และต้องมีการส่งข้อมูลสถานะของระบบเป็นระยะเพื่อปรับข้อมูลในตารางเส้นทางเป็นระยะให้ทันกับการเปลี่ยนแปลงของการสื่อสารในระบบ ซึ่งระยะเวลาในการส่งข้อมูลเพื่อปรับค่าตัวแปรต่าง ๆ นี้จะถูกกำหนดเป็นคาบเวลา (period time) การคำนวณเส้นทางจะพิจารณาเส้นทางที่อยู่ในขอบเขตที่สนใจ แล้วเลือกเส้นทางที่เหมาะสมจากข้อมูลสถานะที่เก็บไว้ในตารางเส้นทาง และเมื่อได้เส้นทางแล้วต้องทำการส่งข้อมูลเส้นทางไปอัพเดทให้เราท์เตอร์หรือสวิตชิ่งโหนดอื่น ๆ ด้วย ซึ่งการหาเส้นทางแบบมีการปรับตัวนี้สามารถแบ่งตามลักษณะการทำงานได้ดังนี้ คือ การหาเส้นทางแบบศูนย์กลาง (Centralized Routing) และการหาเส้นทางแบบกระจาย (Distributed Routing) การหาเส้นทางแบบศูนย์กลางนั้น หากการจัดส่งเป็นเพียงการส่งข้อมูลภายในกลุ่มเดียวกันผู้ส่งจะมองเห็นหรือติดต่อกับผู้รับปลายทางได้โดยตรง แต่ถ้าเป็นการส่งข้อมูลระหว่างโหนดที่เป็นสมาชิกของกลุ่มที่อยู่ติดกันนั้นโหนดในกลุ่มของผู้ส่งต้องมองเห็นและติดต่อกับโหนดในกลุ่มข้างเคียงนั้นได้โดยผ่านทางเราท์เตอร์หลักของแต่ละกลุ่ม กรณีที่โหนดอยู่ในกลุ่มที่ห่างกันจะมีอัลกอริทึมอยู่ 2 แบบที่จะทำให้เราท์เตอร์ของกลุ่มผู้ส่งติดต่อกับเราท์เตอร์ของกลุ่มผู้รับได้ อัลกอริทึมแรกเราท์เตอร์ผู้ส่งเองสามารถมองเห็นเราท์เตอร์ทุกตัวในระบบเครือข่ายหรืออย่างน้อยใน 1 เครือข่ายย่อยและมีข้อมูลสถานะของเราท์เตอร์ทุกตัวเก็บไว้ในตารางเส้นทางเพื่อพิจารณาเลือกเส้นทางที่มีเงื่อนไขการส่งผ่านที่ต้องการ เช่น มีภาระการใช้งานเส้นทางเชื่อมโยงน้อยที่สุด มีแบนด์วิดท์มากที่สุด หรือมีเวลาหน่วงน้อยที่สุดเป็นต้น ทั้งนี้ต้องขึ้นอยู่กับความต้องการในการรับประกันการบริการของแต่ละทราฟฟิกที่ร้องขอเส้นทางเชื่อมโยงและขออนุมัติเส้นทาง ส่วนการหาเส้นทางแบบกระจายนั้นเราท์เตอร์ในกลุ่มผู้ส่งจะทำการเลือกเส้นทางโดยเลือกส่งข้อมูลผ่านไปยังโหนดข้างเคียงตามเงื่อนไขที่ต้องการ เช่น เลือกเส้นทางเชื่อม

โยงที่ไปยังโหนดข้างเคียงที่มีภาระการใช้งานเส้นทางเชื่อมโยงน้อยที่สุดเป็นหลักจากนั้นโหนดข้างเคียงนี้ก็จะเป็นผู้เลือกโหนดถัดไป และให้วิธีการเดียวกันจนกว่าจะส่งข้อมูลไปถึงโหนดของผู้รับปลายทาง การพิจารณาเลือกโหนดถัดไปนั้นจะอาศัยข้อมูลจากโหนดข้างเคียงมาพิจารณาและเป็นข้อมูลเพียงบางส่วนไม่ใช่ข้อมูลที่สมบูรณ์ทั้งหมดเหมือนการหาเส้นทางแบบศูนย์กลาง คุณสมบัติของวิธีการหาเส้นทางทั้งสองแบบเป็นดังนี้

การหาเส้นทางแบบศูนย์กลาง

1. อุปกรณ์เชื่อมเครือข่ายต่างระบบ (gateway) จะต้องรู้ข้อมูลที่สมบูรณ์ในการหาเส้นทางของเราท์เตอร์ทุกตัวในระบบเครือข่ายหรืออย่างน้อยใน 1 เครือข่ายย่อย และเป็นผู้บริหารข้อมูลสภาวะของระบบเครือข่าย

2. อุปกรณ์เชื่อมเครือข่ายต่างระบบจะได้รับข้อมูลทั้งหมดในการการหาเส้นทางที่สมบูรณ์จากโหนดศูนย์กลางของระบบหรือโหนดศูนย์กลางของกลุ่มข้างเคียงเป็นระยะและบันทึกข้อมูลไว้เพื่อทำการพิจารณาหาเส้นทาง ซึ่งเส้นทางที่ได้จะเป็นเส้นทางที่สมบูรณ์จากต้นทางถึงปลายทางหรือเป็นเส้นทางจากเราท์เตอร์ขอบของเครือข่ายต้นทาง (border router) ไปยังเราท์เตอร์ขอบของเครือข่ายปลายทาง เมื่อโหนดศูนย์กลางคำนวณเส้นทางได้แล้วและเก็บข้อมูลเส้นทางลงในตารางเส้นทางแล้วทำการร้องขอเพื่อเชื่อมโยงเส้นทางโดยส่งแพ็กเกตร้องขอเพื่อเชื่อมโยงเส้นทางไปตามเส้นทางที่ได้

การหาเส้นทางแบบกระจาย

1. คอมพิวเตอร์แม่ (Host computer) ทุกตัวและอุปกรณ์เชื่อมเครือข่ายต่างระบบจะมีข้อมูลของเส้นทางย่อยทั้งหมดหรือมีข้อมูลเพียงพอที่จะใช้ในการคำนวณหาเส้นทางไปยังปลายทางหรือไปยังอุปกรณ์เชื่อมโยงในเครือข่าย ซึ่งข้อมูลการใช้งานระบบเครือข่ายและข้อมูลตัวแปรทราฟฟิกจะถูกส่งไปแจ้งยังโหนดต่าง ๆ เพื่อปรับค่าตัวแปรที่จำเป็นต่อการคำนวณหาเส้นทางที่เหมาะสมในแต่ละวิธี

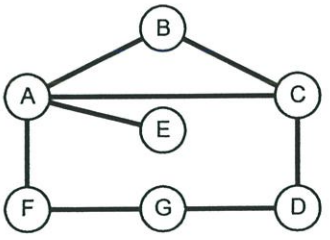
2. อัลกอริทึมที่ใช้ในการหาเส้นทางแบบกระจายนั้นจะต่างกันที่ข้อมูลที่จำเป็นต้องใช้ในกระบวนการหาเส้นทาง วิธีการส่งข้อมูลการใช้งานระบบและตัวแปรทราฟฟิก และวิธีการคำนวณเพื่อหาเส้นทาง ซึ่งจะแบ่งออกเป็นการพิจารณาเวกเตอร์ระยะทาง (Distance vector algorithm) และการพิจารณาสถานะของเส้นทางเชื่อมโยง (Link-State algorithm) ซึ่งการหาเส้นทางแบบเวกเตอร์ระยะทางนั้นอัลกอริทึมที่ใช้งานที่เป็นที่รู้จักจะเป็นแบบเบลแมนฟอร์ดซึ่งข้อมูลที่ส่งนั้นอาจจะเป็นเพียงบางส่วนหรือทั้งหมดแต่จะส่งให้เพียงโหนดข้างเคียงเท่านั้นและข้อมูลที่ส่งเพื่อปรับข้อมูลนั้นจะเป็นข้อมูลขนาดใหญ่ ส่วนวิธีการหาเส้นทางแบบพิจารณาสถานะของเส้นทางเชื่อมโยงนั้น

จะรู้จักดีในอัลกอริทึมการหาเส้นทางที่สั้นที่สุดซึ่งมีการกระจายข้อมูลบางส่วนไปยังทุกโหนดในเครือข่าย โดยแต่ละโหนดจะมีข้อมูลของสถานะการใช้งานเส้นทางเชื่อมโยงที่เชื่อมโยงโดยตรงกับโหนดแล้วส่งข้อมูลดังกล่าวไปยังโหนดต่าง ๆ ดังนั้นเราท์เตอร์แต่ละตัวจึงสามารถสร้างภาพโครงข่ายทั้งหมดได้และในการปรับข้อมูลจะเป็นเพียงข้อมูลบางส่วนเนื่องจากการปรับข้อมูลจะเป็นแบบรวดเร็วเพื่อป้องกันการเกิดการวนลูปในกระบวนการค้นหาเส้นทาง และการหาเส้นทางแบบนี้ยังต้องการใช้หน่วยประมวลผลและหน่วยความจำมากกว่าวิธีการแบบเวกเตอร์ระยะทาง ดังนั้นจึงมีค่าใช้จ่ายในการติดตั้งและการดูแลรักษามากกว่า แต่อัลกอริทึมแบบนี้จะสนับสนุนการรับประกันคุณภาพได้มากกว่า ดังนั้นในงานวิจัยนี้จึงนำวิธีการหาเส้นทางแบบสั้นที่สุดมาปรับปรุงเนื่องจากสามารถส่งข้อมูลสถานะการใช้งานช่องสัญญาณเวลาไปพร้อมกับการปรับข้อมูลสถานะของระบบได้และเส้นทางที่ได้ก็เพื่อให้เป็นไปตามความต้องการในการรับประกันคุณภาพ ซึ่งการหาเส้นทางทั้งสองวิธีนี้จะได้อธิบายรายละเอียดต่อไป

2.1.3. การหาเส้นทางแบบเวกเตอร์ระยะทาง

วิธีการนี้จะบันทึกข้อมูลเส้นทางในตารางเส้นทางซึ่งเป็นข้อมูลที่คำนวณได้จากภาระการใช้งานลิงค์ และแบนด์วิดท์และข้อมูลที่ได้มาจากเราท์เตอร์ที่อยู่ติดกันทุกตัวที่เชื่อมโยงกันอยู่ การเลือกเส้นทางจะเลือกเส้นทางไปยังโหนดที่อยู่ติดกันซึ่งมีภาระการใช้งานเส้นทางเชื่อมโยงน้อยที่สุด และเมื่อเลือกเส้นทางได้แล้วก็จะทำการส่งข้อมูลเส้นทางไปยังเราท์เตอร์ข้างเคียง เมื่อเราท์เตอร์ได้รับข้อมูลเส้นทางใหม่เข้ามา ก็จะทำการคำนวณเส้นทางใหม่และเปลี่ยนข้อมูลเส้นทางในตารางเส้นทาง กรณีที่ต้องทำการคำนวณเส้นทางใหม่จะทำก็ต่อเมื่อได้รับข้อมูลเส้นทางจากเราท์เตอร์ข้างเคียง จากรูปที่ 2.2 จะเห็นได้ว่าในขั้นตอนเริ่มต้นนั้นเราท์เตอร์ A จะทราบเพียงข้อมูลเส้นทางที่จะไปยังโหนดที่อยู่ติดกันเท่านั้น แต่เมื่อโหนดข้างเคียงมีการคำนวณเส้นทางได้แล้วจะทำการปรับข้อมูลเส้นทางและส่งมาแจ้งยังโหนด A ทำให้ A ทราบค่าสถานะการใช้งานเส้นทางเชื่อมโยงและภาระการใช้งานลิงค์ที่จะไปยังโหนดอื่นที่อยู่ไกลออกไปได้จากการปรับข้อมูลตารางเส้นทางของโหนดข้างเคียง และสุดท้ายเมื่อทุกโหนดมีการปรับค่าและส่งข้อมูลระหว่างโหนดข้างเคียงแล้ว จะทำให้โหนด A รู้เส้นทางที่จะไปยังปลายทางทุกโหนดในระบบ แต่การหาเส้นทางในลักษณะนี้มีข้อเสียคืออาจจะเกิดการคำนวณไม่มีที่สิ้นสุด (count to infinity) ได้ กรณีที่จะเกิดการคำนวณไม่มีที่สิ้นสุดนั้นจะเกิดเมื่อมีเส้นทางเชื่อมโยงเสียหายและมีการเชื่อมโยงเส้นทางกับโหนดข้างเคียงอื่นในลักษณะวงปิดดังรูปที่ 2.3 ซึ่งกรณีที่มีเส้นทางใดเส้นทางหนึ่งเสียหาย ซึ่งโหนดที่อยู่ติดกับเส้นทางที่เสียหายจะไม่สามารถไปยังโหนดถัดไปได้และค่าภาระการใช้งานเส้นทางเชื่อมโยงจะถูกเซตค่าเป็นอนันต์แล้วจึงค้นหาเส้นทางอื่นที่มีภาระการใช้งานเส้นทางเชื่อมโยงน้อยที่สุดที่จะไปยังโหนดถัดไปซึ่งในตารางเส้นทางขณะนั้นจะบอกเส้นทางไปยังโหนดที่อยู่ติดกัน แต่โหนดที่อยู่ติดกันนี้ไม่

ทราบเส้นทางที่เสียหายและมีเพียงข้อมูลเส้นทางที่สั้นที่สุดไปยังโหนดที่อยู่ติดกับเส้นทางที่เสียหายจึงส่งข้อมูลและค่าภาระการใช้งานลิงค์กลับไปยังโหนดที่ติดกับเส้นทางที่เสียหาย และทำการส่งข้อมูลกลับไปกลับมาเช่นนี้ทำให้เกิดการคำนวณไม่มีที่สิ้นสุด



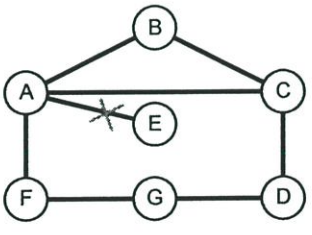
Initial Routing table at A

Destination	Cost	Next hop
B	1	B
C	1	C
D	∞	-
E	1	E
F	1	F
G	∞	-

Final Routing table at A

Destination	Cost	Next hop
B	1	B
C	1	C
D	2	C
E	1	E
F	1	F
G	2	F

รูปที่ 2.2 การหาเส้นทางแบบเวกเตอร์ระยะทาง

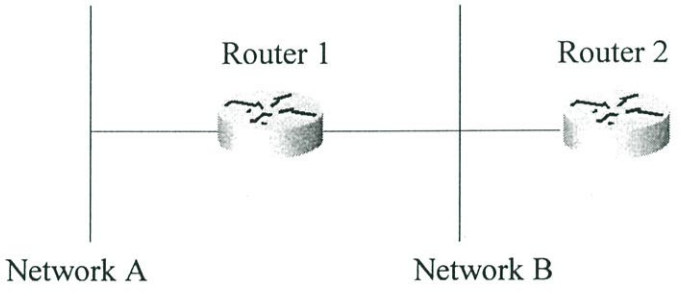


รูปที่ 2.3 การหาเส้นทางแบบเวกเตอร์ระยะทางกรณีมีเส้นทางเสียหายเกิดการคำนวณไม่สิ้นสุด

ในรูปที่ 2.3 เมื่อเส้นทางเชื่อมโยงระหว่างเราท์เตอร์ A และ เราท์เตอร์ E เสียหาย เราท์เตอร์ A จะทำการคำนวณเส้นทางไปยังเราท์เตอร์ E ใหม่ โดยการพิจารณาจากข้อมูลของเราท์เตอร์ B ซึ่งมีค่า ภาระการใช้งานลิงค์จากเราท์เตอร์ B ไปยังเราท์เตอร์ E เท่ากับ 2 ดังนั้น เราท์เตอร์ A จึงคำนวณค่าภาระการใช้งานลิงค์จากเราท์เตอร์ A ไปยังเราท์เตอร์ E ได้เท่ากับ 3 และเลือกใช้เส้นทางผ่านทางเราท์เตอร์ B แล้วปรับค่าภาระการใช้งานลิงค์จากเราท์เตอร์ A ไปยังเราท์เตอร์ E เป็น 3 เมื่อปรับข้อมูลแล้ว เราท์เตอร์ B ก็จะนำข้อมูลของเราท์เตอร์ A มาคำนวณและปรับค่าภาระ

การใช้งานลิงค์จากเราท์เตอร์ B ไปยังเราท์เตอร์ E ใหม่เป็น 4 เมื่อเราท์เตอร์ B ปรับค่าใหม่แล้ว เราท์เตอร์ A ก็จะคำนวณใหม่เป็นค่าภาระการใช้งานลิงค์ เท่ากับ 5 ซึ่งการคำนวณลักษณะนี้จะเพิ่มค่าภาระลิงค์ขึ้นไปได้เรื่อย ๆ เกิดลักษณะของการคำนวณและปรับค่าไม่มีที่สิ้นสุด ซึ่งกรณีที่เกิดปัญหาลักษณะนี้จะสามารถแก้ไขได้โดยใช้วิธีการดังต่อไปนี้

1. Split Horizon คือการไม่ส่งข้อมูลไปกลับไปยังเส้นทางที่รับข้อมูลมา ตัวอย่างในรูปที่ 2.4 เราท์เตอร์ 1 ประกาศออกไปว่าเชื่อมโยงอยู่กับเครือข่าย A ดังนั้นจึงไม่มีเหตุผลที่เราท์เตอร์ 2 จะต้องส่งข้อมูลที่ปรับกลับไปยังเราท์เตอร์ A เพราะเราท์เตอร์ A นั้นเชื่อมโยงอยู่กับเครือข่าย A โดยตรง และในการประกาศของเราท์เตอร์ 2 จะบอกด้วยว่าไม่ต้องส่งกลับไปยังเราท์เตอร์ 1 และเป็นการป้องกันการแจ้งข้อมูลย้อนกลับกรณีที่การเชื่อมโยงระหว่างเราท์เตอร์ 1 กับเครือข่าย A เสียหาย เพราะหากเส้นทางเชื่อมโยงดังกล่าวเสียหายเราท์เตอร์ 1 จะใช้ข้อมูลของเราท์เตอร์ 2 ที่บอกว่าจะไปยังเครือข่าย A ได้ แล้วเราท์เตอร์ 1 ก็จะนำข้อมูลมาปรับว่าการจะไปทางเครือข่าย A จะต้องผ่านทางเราท์เตอร์ 2 ซึ่งกรณีเช่นนี้จะทำให้เกิดการคำนวณแบบไม่มีสิ้นสุดได้



รูปที่ 2.4 ตัวอย่างวิธีการ Split-Horizon

2. Hold-down Time เป็นการกำหนดให้เราท์เตอร์ไม่รับข้อมูลในช่วงเวลาที่เร็วเกินไป เมื่อมีเราท์เตอร์เสียหาย เราท์เตอร์ข้างเคียงจะรู้ได้จากเวลารอคอยแพ็กเกตเพื่อปรับค่าข้อมูลเกินจากเวลาที่กำหนด และเราท์เตอร์ข้างเคียงนี้จะทำการคำนวณเส้นทางใหม่และส่งข้อมูลไปปรับตารางเส้นทางที่เราท์เตอร์อื่น ๆ ซึ่งเวลาที่ส่งข้อมูลในการปรับตารางเส้นทางนั้นเคลื่อนไปจากเวลาปกติ เราท์เตอร์จึงตั้งเวลาในการรอโดยไม่ทำการปรับค่าใด ๆ ในช่วงเวลาดังกล่าว ซึ่งเวลารอคอยจะถูกคำนวณให้มีช่วงเวลานานกว่าเวลาในการปรับข้อมูล

3. Poison Reverse เป็นการกำหนดค่าจำนวนจุดเชื่อมโยงสูงสุดเอาไว้ เมื่อจำนวนจุดเชื่อมโยงในเส้นทางเกินที่กำหนดจะทำให้เส้นทางไปยังโหนดดังกล่าวถูกกำหนดให้เป็นอนันต์ซึ่งเป็นเส้นทางที่ไม่สามารถใช้งานได้ เป็นเหมือนการบอกให้เอาเส้นทางดังกล่าวออกจากตารางเส้นทางและแทนค่าดังกล่าวนี้ในช่วงเวลา hold down time

ข้อดีของวิธีการหาเส้นทางแบบเวกเตอร์ระยะทางนี้คือ ง่ายต่อการปรับปรุงและนำไปใช้งานจริงมีการใช้งานอย่างกว้างขวาง ยืดหยุ่นในการติดตั้งระบบและเปลี่ยนแปลงตามเงื่อนไข ตัวอย่างของการหาเส้นทางแบบเวกเตอร์ระยะทางได้แก่ Routing Information Protocol (RIP) ซึ่งการหาเส้นทางวิธีนี้เป็นการหาเส้นทางแบบเวกเตอร์ระยะทาง จะมีการส่งข้อมูลเพื่อปรับค่าข้อมูลในการหาเส้นทางเป็นระยะโดยปกติจะถูกกำหนดไว้ที่ทุก30 วินาที เมื่อเราท์เตอร์ได้รับข้อมูลใหม่เข้ามาจะทำการปรับข้อมูลเส้นทางในตารางเส้นทางทันทีและเมื่อปรับเส้นทางแล้วจะต้องแจ้งข้อมูลไปยังเราท์เตอร์ข้างเคียงทันที การพิจารณาเลือกเส้นทางจะพิจารณาจากจำนวนจุดเชื่อมโยงระหว่างโหนดต้นทางไปยังปลายทางเป็นหลัก โดยกำหนดค่าสูงสุดที่ 15 จุดเชื่อมโยง ดังนั้นหากเส้นทางนั้นมีจุดเชื่อมโยงที่ 16 จะถือว่าไม่สามารถใช้เส้นทางนั้นได้ ในทางกลับกันเส้นทางที่มีจำนวนจุดเชื่อมโยงน้อยที่สุดถือเป็นเส้นทางที่เหมาะสมที่สุดในกรณีนี้ การหาเส้นทางแบบนี้เหมาะกับการเชื่อมโยงเครือข่ายที่มีขนาดไม่ใหญ่มากและไม่ซับซ้อน ในปัจจุบันมีการใช้งานโปรโตคอล RIP อย่างแพร่หลายเพราะสามารถนำไปใช้งานได้ง่าย และมีอุปกรณ์เราท์เตอร์แทบทุกยี่ห้อรองรับ

2.1.4. การหาเส้นทางแบบพิจารณาสถานะการใช้งานเส้นทางเชื่อมโยง

ลักษณะการทำงานแบบพิจารณาสถานะการใช้งานเส้นทางเชื่อมโยงนั้นจะมีการส่งข้อมูลการเชื่อมโยงจากเราท์เตอร์แต่ละตัวไปยังเราท์เตอร์อื่น ๆ ในระบบ ซึ่งข้อมูลนี้เป็นสถานะการใช้งานเส้นทางเชื่อมโยงในระบบ ข้อมูลสถานะการใช้งานเส้นทางเชื่อมโยงที่ส่งออกไปในเครือข่ายของเราท์เตอร์แต่ละตัวจะเป็นข้อมูลที่บอกว่าเราท์เตอร์มีการเชื่อมต่ออยู่กับเครือข่ายใดอย่างไรและในกรณีที่มีการเปลี่ยนแปลงภายในเครือข่าย เช่น มีบางวงจรล่มไปก็จะมีการส่งข้อมูลเฉพาะที่มีการเปลี่ยนแปลงไปซึ่งมีขนาดไม่ใหญ่มาก ในการคำนวณเพื่อเลือกเส้นทางนั้นเราท์เตอร์จะพิจารณาจากข้อมูลสถานะการเชื่อมโยงที่มีทั้งหมดแล้วเก็บเส้นทางที่เลือกได้ไว้ในตารางเส้นทาง ตัวอย่างโปรโตคอลที่ใช้กลไกการหาเส้นทางแบบพิจารณาสถานะการใช้งานเส้นทางเชื่อมโยงได้แก่ การหาเส้นทางแบบสั้นที่สุดซึ่งข้อมูลสถานะการใช้งานเส้นทางเชื่อมโยงนั้นจะถูกส่งไปพร้อมกับแพ็กเกตแจ้งสถานะการใช้งานเส้นทางเชื่อมโยงซึ่งกระจายไปยังเราท์เตอร์ทุกโหนดเป็นระยะ ข้อมูลที่ส่งมานี้จะถูกนำมาพิจารณาในการคำนวณหาเส้นทาง ซึ่งแพ็กเกตแจ้งสถานะการใช้งานเส้นทางเชื่อมโยง ประกอบด้วย

1. หมายเลขของสวิตชิ่งโหนดที่สร้างแพ็กเกตแจ้งสถานะการใช้งานเส้นทางเชื่อมโยง(ID)
2. ลำดับหมายเลขลำดับของแพ็กเกต (sequence number)
3. เวลาที่แพ็กเกตจะอยู่ในระบบ (time-to-live, TTL)
4. ข้อมูลสำคัญในการพิจารณาหาเส้นทาง เช่น ค่าภาระการใช้งานลิงค์

ตัวอย่างการหาเส้นทางแบบนี้ได้แก่วิธีการหาเส้นทางที่สั้นที่ ซึ่งเป็นโปรโตคอลที่ใช้วิธีพิจารณาเส้นทางและปรับปรุงสถานะแบบพิจารณาสถานะการใช้งานเส้นทางเชื่อมโยง โดยมีการพัฒนาขึ้นในราวปี ค.ศ. 1989 มีคุณสมบัติเด่นคือใช้ทรัพยากรในเครื่องไม่มาก ทำให้บริษัทที่เป็นผู้ให้บริการอินเทอร์เนตและเครือข่ายต่าง ๆ ที่เชื่อมต่อในอินเตอร์เน็ตหันมาใช้โปรโตคอลการหาเส้นทางแบบสั้นที่สุดเพื่อเชื่อมต่อภายในเครือข่ายตนเองกันมากขึ้น และสามารถรองรับกับเครือข่ายขนาดใหญ่ได้ดีนอกจากนี้การหาเส้นทางแบบสั้นที่สุดยังสามารถรองรับการกำหนดเครือข่ายย่อยได้อย่างมีประสิทธิภาพอีกด้วย ซึ่งช่วยให้ผู้จัดการเครือข่ายสามารถกำหนดแยกเครือข่ายออกเป็นเครือข่ายย่อย ๆ ได้หลายรูปแบบ ในการแลกเปลี่ยนข้อมูลและสถานะของเครือข่ายและข้อมูลในตารางเส้นทางนั้น จะติดต่อแลกเปลี่ยนข้อมูลระหว่างเราท์เตอร์ที่อยู่ในระบบเครือข่ายเดียวกันคือระหว่างเราท์เตอร์หลัก (Autonomous system router) และเราท์เตอร์ บริวารที่เชื่อมโยงกันอยู่ นอกจากนี้ยังสามารถแลกเปลี่ยนข้อมูลกับเราท์เตอร์ที่ใช้โปรโตคอลอื่นได้อีกด้วยโดยอาศัยกลไกการอ้างอิงของเราท์เตอร์หลัก อาจกล่าวได้ว่าการหาเส้นทางแบบสั้นที่สุดเป็นการหาเส้นทางแบบนี้เป็นแบบที่พิจารณาค่าภาระการใช้งานลิงค์ เป็นการหาเส้นทางที่หลากเส้นทาง (multipath routing) และ การหาเส้นทางตามความต้องการของการให้บริการ (type-of-service, TOS) ซึ่งจะสนับสนุนหรือรองรับงานประยุกต์จากชั้นการประยุกต์ เช่น การประยุกต์ใช้งานกับอินเทอร์เนตโพรโทคอล (IP TOS) จะใช้ข้อมูลเวลาหน่วง ปริมาณทราฟฟิกที่ใช้ในเส้นทางเชื่อมโยงและความน่าเชื่อถือในการพิจารณา และจะเลือกเส้นทางที่มีเวลาหน่วงน้อย มีทราฟฟิกในเส้นทางเชื่อมโยงน้อยแต่มีความน่าเชื่อถือสูง เป็นต้น วิธีการหาเส้นทางแบบสั้นที่สุดนี้ได้นำอัลกอริทึมการหาเส้นทางที่สั้นที่สุด Dijkstra's shortest path algorithm [19] มาใช้ซึ่งจะอธิบายการทำงานของอัลกอริทึมได้ดังนี้

เริ่มอัลกอริทึมโดยเซตของโหนดในเส้นทางจะเก็บค่าโหนดต้นทาง A แล้วจึงพิจารณาโหนดปลายทาง v ทุกโหนดที่มี หากโหนด v เชื่อมโยงกับโหนด A โดยตรงจะเก็บค่าภาระการใช้งานลิงค์ $c(A, v)$ ไว้ที่ $D(v)$ แต่หากไม่ได้เชื่อมโยงโดยตรงจะเก็บค่า ∞ จากนั้นทำการตรวจสอบโหนดอื่น ๆ ที่ไม่ได้อยู่ในเซตของโหนดในเส้นทาง w เมื่อเปรียบเทียบค่าภาระการใช้งานลิงค์จาก A ถึง w หรือ $D(w)$ ทุกโหนด w แล้วจะเลือกเก็บโหนดที่ค่าภาระการใช้งานลิงค์ $D(w)$ น้อยที่สุดเก็บลงในเซต N แล้วพิจารณาโหนดปลายทาง v ที่เหลือทุกโหนดที่ไม่ได้อยู่ใน N แล้วเก็บค่าภาระการใช้งานลิงค์ไปยังโหนดปลายทางดังกล่าวได้เป็นค่าภาระการใช้งานลิงค์ที่น้อยที่สุดในขณะนั้นเป็น $D(w)$ รวมกับค่าภาระการใช้งานลิงค์จากโหนด w ไปยังโหนด v และทำในลักษณะเดียวกันนี้สำหรับทุกโหนดที่เหลืออยู่ก็จะได้เส้นทางที่มีค่าภาระการใช้งานลิงค์น้อยที่สุด หรือเป็นเส้นทางสั้นที่สุดไปยังปลายทางใด ๆ ในเครือข่าย

Initialization:

$N = \{A\}$

for all nodes v

if v adjacent to A then $D(v) = c(A, v)$

else $D(v) = \infty$

Loop

Find w not in N such that $D(w)$ is a minimum

Add w to N

Update $D(v)$ for all v adjacent to w and not in N:

$D(v) = min\ (D(v), D(w) + c(w,v))$

/*new cost to v is either old cost to v or known shortest path cost to w plus cost from w to v */

until all nodes in N

$c(i, j)$ เป็นค่าภาระการใช้งานลิงค์ของเส้นทางเชื่อมโยงระหว่างโหนด i กับโหนด j ถ้าโหนด i กับโหนด j ไม่ได้ต่อกันโดยตรงแล้วค่า $c(i, j) = \infty$ และกำหนดให้ $c(i, j) = c(j, i)$

$D(v)$ เป็นค่าภาระการใช้งานลิงค์ที่น้อยที่สุดจากโหนดต้นทางไปยังโหนด v

$p(v)$ เป็นโหนดก่อนหน้าโหนด v (จากโหนดต้นทางไปยังโหนด v)

N เป็นเซตของโหนดในเส้นทางซึ่งมีค่าภาระการใช้งานลิงค์น้อยที่สุด

2.2 การค้นหาเส้นทางกับการรับประกันคุณภาพการให้บริการ

การประยุกต์ใช้งานในระบบสื่อสารข้อมูลต่างก็ต้องการการรับประกันคุณภาพในการให้บริการที่ต่างกัน เช่น การพิจารณาเวลาหน่วง การแกว่งของเวลาหน่วง แบนด์วิดท์ และอัตราการสูญหายของแพ็กเกต ซึ่งตัวแปรควบคุมเหล่านี้มักจะถูกคำนวณในรูปของภาระการใช้งานเส้นทางเชื่อมโยง การรับประกันคุณภาพการบริการจะพัฒนาเพื่อสนับสนุนการสื่อสารข้อมูล ที่ต้องการการรับประกันคุณภาพดังที่กล่าวมาและสนับสนุนการสื่อสารที่ไม่มีการรับประกันคุณภาพการบริการ (Best-effort traffic) ซึ่งการที่จะรับประกันคุณภาพการบริการนั้นจะต้องพิจารณาเฉพาะสำหรับแต่ละทราฟฟิกและต้องลดปัญหาเรื่องการปฏิเสธการร้องขอเพื่อเชื่อมโยงเส้นทางของแต่ละทราฟฟิก ส่วนทราฟฟิกที่ไม่ได้ต้องการการรับประกันคุณภาพการบริการนั้น จะพิจารณาประสิทธิภาพของการรับส่งแพ็กเกตข้อมูลโดยรวม และเวลาที่ใช้ในการรับส่งข้อมูลเป็นสำคัญ ในการคำนวนเพื่อหาเส้นทางที่มีการรับประกันคุณภาพจำเป็นต้องรู้ข้อมูลสถานะของระบบ

(information of network state) ซึ่งจะประกอบด้วยข้อมูลโครงข่ายของระบบ (Topology) ข้อมูลสถานะการใช้งานเส้นทางเชื่อมโยง ตัวอย่างข้อมูลสถานะการใช้งานเส้นทางเชื่อมโยง เช่น แบนด์วิดท์ที่สามารถใช้งานและเวลาหน่วงของการส่งผ่านเส้นทางเชื่อมโยงเป็นต้น ซึ่งวิธีการหาเส้นทางนั้นมีอยู่หลายวิธีและแต่ละวิธีต่างก็มีการคำนวณตัวแปรทราฟฟิกที่ต่างกันออกไป และ การพิจารณาตัวแปรทราฟฟิกจะพิจารณาจากอัตราการปฏิเสธการร้องขอเส้นทางเชื่อมโยงและ ขออนุมัติเส้นทาง ซึ่งจะบอกถึงเปอร์เซ็นต์ของการถูกปฏิเสธที่จะอนุมัติเส้นทางหรือบอกถึงจำนวนของการร้องขอเส้นทางที่ไม่ได้รับการยอมรับจากระบบ ค่าต่ำสุดของอัตราการปฏิเสธการร้องขอเส้นทางนี้จะขึ้นอยู่กับการทำงานของระบบและรูปแบบการกระจายของทราฟฟิกที่ร้องขออนุมัติเพื่อใช้งานในระบบ สาเหตุของการปฏิเสธการร้องขอเส้นทางจะเกิดขึ้นเนื่องจากไม่มีทรัพยากรหรือแบนด์วิดท์ที่เพียงพอที่จะให้บริการแก่ทราฟฟิกที่ร้องขอ ดังนั้นอัลกอริทึมการหาเส้นทางต่าง ๆ จึงมีการพัฒนาเพื่อที่จะลดอัตราการถูกปฏิเสธการร้องขอเพื่ออนุมัติเส้นทางให้อยู่ในช่วงค่าต่ำสุดที่ยอมรับได้ให้มากที่สุด

คุณสมบัติที่จำเป็นในการพิจารณาเพื่ออนุมัติเส้นทางหรือปฏิเสธเส้นทางจะประกอบด้วย การสิ้นเปลืองแบนด์วิดท์ในกระบวนการสร้างเส้นทาง (Network Overhead) ซึ่งจะเป็นการแบ่งแบนด์วิดท์ในระบบส่วนหนึ่งมาใช้สำหรับกระบวนการแลกเปลี่ยนข้อมูลเส้นทางทั้งนี้ขึ้นอยู่กับจำนวนและขนาดของข้อมูล ความถี่ในการปรับปรุงข้อมูล วิธีการหาเส้นทางที่ใช้รวมทั้งโหลดในการคำนวณของเราท์เตอร์แต่ละตัว สิ่งที่จะพิจารณาถัดมาคือเวลาที่ใช้ในกระบวนการร้องขอเพื่อเชื่อมโยงเส้นทางและการอนุมัติเส้นทาง (Call Setup Time) เป็นเวลาตั้งแต่ที่เราท์เตอร์ได้รับแพ็กเกตร้องขอการเชื่อมโยงเส้นทางและส่งออกไปจนกระทั่งทำการสร้างเส้นทางเสร็จเรียบร้อย การพิจารณาถัดมาจะเป็นค่าภาระการใช้งานเส้นทางเชื่อมโยง (Source Computation Cost) ซึ่งเราท์เตอร์จะทำการคำนวณหลังจากที่ได้รับข้อมูลตัวแปรทราฟฟิกที่ร้องขอการรับประกันการบริการ สิ่งที่จะพิจารณาต่อมาคือ ขนาดของระบบ (Scalability) ซึ่งจะขึ้นอยู่กับการทำงานของอัลกอริทึมการหาเส้นทางแต่ละแบบ เมื่อมีการเพิ่มขนาดของระบบเครือข่าย หรือมีอุปกรณ์เชื่อมต่อในระบบเพิ่มขึ้น การรู้ข้อมูลที่ถูกต้องล่าสุดของระบบจะช่วยลดอัตราการปฏิเสธการร้องขอเส้นทางได้ แต่การส่งข้อมูลสถานะของระบบที่กระจายออกมาในระบบเครือข่ายอย่างต่อเนื่องนั้นจะเพิ่มมากขึ้นด้วย ทำให้เกิดปัญหาสิ้นเปลืองทรัพยากรของระบบมากเกินไป ความจำเป็น ที่ต้องมีการส่งข้อมูลสถานะของระบบเป็นระยะเนื่องจากข้อมูลที่จะนำมาใช้ในการตัดสินใจเลือกเส้นทางควรจะเป็นข้อมูลที่ใหม่อยู่ตลอดเวลา เพราะการนำข้อมูลเก่ามาใช้ในการคำนวณอาจจะทำให้ได้เส้นทางที่ไม่ดี เป็นการเพิ่มการปฏิเสธการร้องขอการเชื่อมโยงเส้นทาง แต่หากปรับปรุงข้อมูลอยู่ตลอดเวลาจะได้ข้อมูลที่เป็นปัจจุบัน ทำให้การคำนวณหาเส้นทางมีความแม่นยำและมีประสิทธิภาพมากขึ้นสามารถเลือกเส้นทางที่ดีได้ ขั้นตอนการหาเส้นทางนั้นมีอยู่ 2 ขั้นตอน

ขั้นตอนแรกเป็นการเก็บข้อมูลสถานะของระบบ ไว้ในแต่ละโหนดที่อยู่ในระบบเครือข่าย ขั้นตอนที่สองเป็นการนำข้อมูลที่ได้มาพิจารณาเส้นทางที่เป็นไปได้และทำการคำนวณหาเส้นทางที่เหมาะสมที่สุดตามความต้องการในการรับประกันการบริการ ซึ่งมีความจำเป็นต้องรู้ข้อมูลสถานะการใช้งานเส้นทางเชื่อมโยงทุกเส้นทางเชื่อมโยงที่เชื่อมโยงอยู่ตั้งแต่โหนดต้นทางถึงโหนดปลายทาง และการพิจารณาหาเส้นทางที่เป็นไปได้นี้จะขึ้นอยู่กับวิธีการเก็บข้อมูลสถานะและตำแหน่งที่ทำการเก็บข้อมูลสถานะไว้ในระบบ ซึ่งแบ่งวิธีการหาเส้นทางตามลักษณะการเก็บข้อมูลของระบบนี้ออกเป็น 3 วิธี คือ

การหาเส้นทางที่เราท์เตอร์ต้นทาง (Source routing) การหาเส้นทางแบบกระจาย (Distributed routing) และการหาเส้นทางแบบลำดับชั้น (Hierarchical routing) ซึ่งวิธีการหาเส้นทางที่เราท์เตอร์ต้นทางนั้นเราท์เตอร์แต่ละตัวจะเก็บข้อมูลสถานะของทั้งระบบ (complete global state) ซึ่งประกอบด้วย รูปแบบโครงข่ายของระบบและข้อมูลสถานะของเส้นทางเชื่อมโยงทั้งหมด เนื่องจากข้อมูลที่มีเป็นข้อมูลของทั้งระบบ ดังนั้นเส้นทางที่เป็นไปได้ทั้งหมดจึงสามารถคำนวณได้ที่เราท์เตอร์ต้นทาง แล้วจึงทำการส่งแพ็กเกตร้องขอการเชื่อมโยงเส้นทางออกไปตามเส้นทางที่เลือกเพื่อทำการขออนุมัติและเชื่อมโยงเส้นทางต่อไป ตัวอย่างของการหาเส้นทางแบบนี้ได้แก่ การหาเส้นทางแบบพิจารณาสถานะการใช้งานเส้นทางเชื่อมโยง ซึ่งจะทำการส่งข้อมูลสถานะของทั้งระบบให้กับเราท์เตอร์ทุกตัว การกระจายข้อมูลสถานะทั้งหมดไปยังเราท์เตอร์ทุกตัวในกลุ่มและตัดสินใจเลือกเส้นทางที่เราท์เตอร์ต้นทางได้ ข้อมูลของทั้งระบบถูกเก็บไว้ในทุกโหนดซึ่งจะต้องมีการปรับปรุงข้อมูลเป็นระยะเพราะข้อมูลการใช้งานของระบบมีการเปลี่ยนแปลงตลอดเวลา ต้องใช้วิธีการหาเส้นทางแบบพิจารณาสถานะการใช้งานเส้นทางเชื่อมโยงในการประมาณค่าข้อมูลจนกว่าจะได้รับแพ็กเกตแจ้งสถานะการใช้งานเส้นทางเชื่อมโยงที่ส่งมาใหม่ ซึ่งความแม่นยำจะน้อยลงและมีผลต่อการคำนวณการรับประกันคุณภาพการให้บริการ การอธิบายผลกระทบของการคำนวณหาเส้นทางเนื่องจากการใช้ข้อมูลเก่าในการเลือกเส้นทาง ต้องทำการสุ่มการกระจายของโหลดที่ใช้ในขั้นตอนการหาเส้นทางและพิจารณาเรื่องของค่าโสหุ้ยสูญเปล่า (overhead) หรือส่วนเกินที่จำเป็นที่เกิดขึ้นในการหาเส้นทางที่เราท์เตอร์ต้นทางการคำนวณค่าโสหุ้ยสูญเปล่า โดยเฉพาะกรณีที่มีการเพิ่มขนาดของระบบโดยรวดเร็วและเป็นระบบที่หลากหลาย ดังนั้นการคำนวณหาเส้นทางต้องคำนวณเพื่อให้รับประกันคุณภาพการบริการมากขึ้น การหาเส้นทางสำหรับแต่ละทราฟฟิกที่เพิ่มขึ้นมีหลายวิธีการที่มีการนำเสนอสำหรับ unicast source routing และ multicast source routing ซึ่งทั้งสองวิธีต่างต้องมีการเก็บข้อมูลของทั้งระบบไว้ที่ทุก ๆ โหนด ซึ่ง อัลกอริทึมส่วนใหญ่สำหรับแบบ unicast นี้มักจะใช้วิธีการหาเส้นทางที่สั้นที่สุด เช่น วิธีการหาเส้นทางแบบ Djikstra วิธีการที่ใช้กันมากคือการปรับปรุงรับประกันคุณภาพในการเส้นทางแบบ OSPF หรือที่เรียกว่า Q-OSPF ประกอบด้วยวิธีการหาเส้นทางที่เราท์เตอร์ต้นทาง แต่จะมีการกำหนดค่าแบนด์

วิดท์และโหลดที่เกิดขึ้นในกระบวนการต่าง ๆ ในระบบเครือข่าย ดังนั้นสวิตช์ทุกตัวจะต้องเก็บข้อมูลของตัวเองไว้ว่ามีทรัพยากรเหลืออยู่เท่าไหร่ มีการส่งข้อมูลสถานะการใช้งานเส้นทางเชื่อมโยงไปยังสวิตช์ตัวอื่น และสุดท้ายก็จะเป็นกระบวนการร้องขอเพื่อเชื่อมโยงเส้นทางของทุก ๆ เส้นทาง

ส่วนวิธีการหาเส้นทางแบบกระจายนั้นจะถูกคำนวณที่แต่ละโหนดขณะที่มีการส่งข้อมูลและเปลี่ยนแปลงค่าภาระการใช้งานเส้นทางเชื่อมโยงระหว่างโหนดข้างเคียง ข้อมูลที่แลกเปลี่ยนกันของแต่ละโหนดนี้จะถูกนำมาใช้ในการพิจารณาเพื่อเลือกโหนดถัดไปในเส้นทาง การหาเส้นทางแบบกระจายส่วนใหญ่จะใช้วิธีการคำนวณแบบเวกเตอร์ระยะทาง ซึ่งจะกระทำที่สวิตชิ่งโหนดแต่ละตัว โดยที่สวิตชิ่งโหนดแต่ละตัวจะทำการพิจารณาเพื่อเลือกโหนดถัดไปในเส้นทาง อาจเรียกได้ว่าเป็นการหาเส้นทางแบบจุดต่อจุด

ส่วนวิธีการหาเส้นทางแบบลำดับชั้นนั้นสวิตชิ่งโหนดจะถูกแบ่งออกเป็นกลุ่ม ในแต่ละกลุ่มจะแบ่งเป็นลำดับชั้นย่อยลงไปอีก ซึ่งสวิตชิ่งโหนดในลำดับชั้นบนสุดของแต่ละกลุ่มจะมีข้อมูลสถานะของการใช้งานเส้นทางเชื่อมโยงและข้อมูลการเชื่อมโยงระหว่างสวิตชิ่งโหนดที่เป็นสมาชิกในแต่ละลำดับชั้นภายในกลุ่ม และมีข้อมูลรวมทั้งระบบ (Aggregate state information) ซึ่งจะเป็นข้อมูลของกลุ่มอื่น ๆ วิธีการหาเส้นทางจะเริ่มต้นที่สวิตชิ่งโหนดต้นทางคำนวณเส้นทางเพื่อไปยังสวิตช์หรือเราท์เตอร์ที่อยู่ลำดับชั้นบนสุดแล้วส่งแพ็กเกตร้องขออนุมัติการใช้งานเส้นทางเชื่อมโยงไปยังเราท์เตอร์ในลำดับชั้นบนสุดของกลุ่มและเราท์ตอร์ในลำดับชั้นบนสุดนี้จะทำการตรวจสอบที่อยู่ของโหนดปลายทางว่าอยู่ในกลุ่มใดแล้วจึงพิจารณาตารางเส้นทางว่าจะใช้เส้นทางใดเพื่อไปยังปลายทางที่อยู่ในกลุ่มดังกล่าว แล้วจึงทำการส่งแพ็กเกตร้องขออนุมัติการใช้งานเส้นทางเชื่อมโยง

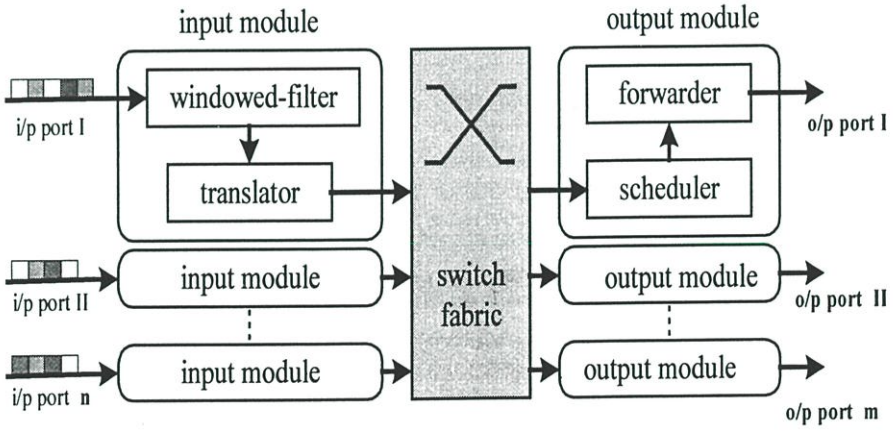
ในวิธีการของ Djikstra นั้น จะพิจารณาค่าภาระการใช้งานลิงค์ตั้งแต่โหนดต้นทางถึงโหนดปลายทางใด ๆ ในทุกเส้นทางที่เป็นไปได้ แล้วเลือกเส้นทางที่ค่าภาระการใช้งานลิงค์รวมในเส้นทางน้อยที่สุดเพื่อหาเส้นทางที่สั้นที่สุด จะเห็นว่าสิ่งที่นำมาพิจารณามีเพียงข้อมูลค่าภาระการใช้งานลิงค์เท่านั้น หากต้องการให้การหาเส้นทางรับรองการรับประกันการบริการด้วยจำเป็นจะต้องนำพารามิเตอร์อื่นมาพิจารณาร่วม วิธีการที่ใช้ในงานวิจัยนี้คือการนำชุดข้อมูลสถานะการใช้งานช่องสัญญาณเวลามาพิจารณาเนื่องจากข้อมูลสถานะการใช้งานช่องสัญญาณเวลาบอกถึงแบนด์วิดท์ที่สามารถใช้งานได้ในระบบและยังบอกถึงจำนวนตำแหน่งที่ซิงโครนัสระหว่างโหนดในเส้นทางด้วย และเมื่อนำไปคำนวณร่วมกับเวลาหน่วงก็จะได้ค่าภาระการใช้งานลิงค์ที่เหมาะสมสำหรับการสื่อสารที่มีการกำหนดช่องสัญญาณเวลาในเฟรมขนาดคงที่ได้และหากต้องการเพิ่มการรับประกันด้านอื่นก็สามารถนำตัวแปรควบคุมต่าง ๆ มาพิจารณาร่วมเป็นฟังก์ชันของค่าภาระการใช้งานลิงค์ได้ หรืออาจจะนำมากำหนดเป็นขอบเขตเพื่อควบคุมคุณสมบัติของเส้นทางด้านนั้นให้เป็นไปตามความต้องการใช้งานได้ และเมื่อได้เส้นทางที่เหมาะสมกับความต้องการแล้วจะต้อง

พิจารณาในเรื่องของการร้องขอเพื่อเชื่อมโยงเส้นทางและการจัดลำดับแพ็กเกตเพื่อส่งออกซึ่งจะอธิบายต่อไปในหัวข้อ 2.3

2.3 หลักการ W-TDM

การจัดลำดับแพ็คเก็ตเพื่อส่งออกแบบ W-TDM นั้นจะถูกพัฒนาลงในกระบวนการทำงานของสวิตช์ การจัดลำดับแพ็กเก็ตเพื่อส่งออกเป็นการทำงานของสวิตช์ซึ่งโหนดในส่วนที่ทำการจัดเรียงลำดับชุดข้อมูลแพ็กเกตและจัดลำดับการส่งออกไปยังทางออก (output port) ซึ่งในหลักการ W-TDM จะใช้การจัดเรียงข้อมูลตามลำดับความสำคัญ (sorted-priority) และ ใช้การจัดการเฟรม (frame-based strategy)

ส่วนที่เป็นการจัดเรียงตามลำดับความสำคัญนั้นจะมีตัวแปรที่อ้างอิงถึงเวลาที่สัมพันธ์กับเส้นทางเชื่อมโยงขาออก (outgoing-link) ของแต่ละสวิตช์ และแพ็กเกตที่ได้รับเข้ามาก็จะมีการบันทึกเวลาเข้าตามจริง ส่วนที่เป็นการจัดลำดับแพ็คเก็ตเพื่อส่งออกจะทำการแยกและจัดเรียงแพ็กเกตในบัฟเฟอร์ขาออก (forwarding output buffer) ของสวิตช์ โดยสัมพันธ์กับเวลาการเข้ามาที่บันทึกไว้และส่งผ่านไปยังบัฟเฟอร์ขาออกที่จองไว้ โครงสร้างภายในสวิตช์โหนดแบ่งออกเป็นสามส่วนดังรูปที่ 2.5 คือ หน่วยด้านเข้า หน่วยด้านออกและส่วนสวิตช์ (switching fabric) ซึ่งหลักการ W-TDM จะใช้ space-switch เพื่อสวิตช์แพ็กเกตจากต้นทางไปยังปลายทาง



รูปที่ 2.5 โครงสร้างของสวิตช์ในหลักการของ W-TDM

องค์ประกอบต่างๆ ภายในสวิตช์ ซึ่งใช้หลักการ W-TDM

1. Windowed-filter มีหน้าที่กรองแพ็กเกตที่รับเข้ามาในส่วนอินพุทโดยใช้หลักการของ วินโดว์
2. Translator มีหน้าที่แปลและเปลี่ยนข้อมูลที่ส่วนหัวของแพ็กเกต (packet header) ที่เข้ามากับข้อมูลในตารางการจัดลำดับแพ็กเกตเพื่อส่งออก ซึ่งข้อมูลนี้ได้จากกระบวนการร้องขออนุมัติใช้งานเส้นทางเชื่อมโยง ซึ่งจะทำการเปรียบเทียบ ค่า Virtual circuit identifier (VCI) จากแพ็กเกต

ที่ได้รับเข้ามากับค่า VCI ในตาราง เมื่อค่า VCI ตรงกันแล้วจะเปลี่ยนค่า VCI ของแพ็กเกตเป็นค่าใหม่ ซึ่งค่า VCI ใหม่นี้จะถูกนำไปใช้ในขั้นตอนการจัดลำดับแพ็คเก็ตเพื่อส่งออกในสวิตซิ่งโหนดถัดไป และเมื่อทำการแปลและเปลี่ยนส่วนหัวของแพ็กเกตข้อมูลแล้วจะต้องทำการคำนวนหาตำแหน่งช่องสัญญาณเวลาที่บัฟเฟอร์ขาออกซึ่งถูกสำรองไว้ให้ที่ทางออกด้วยเพื่อนำแพ็กเกตไปเก็บในตำแหน่งที่สัมพันธ์กับการซิงโครนัสของการรับส่งแพ็กเกต

3. Scheduler ขอบเขตการให้บริการของ scheduler จะถูกกำหนดโดย วินโดว์ นั่นคือ วินโดว์ที่ทางเข้า (input port) ทุกทางจะต้องซิงโครนัสกับวินโดว์ที่ทางออก ของ scheduler ทุกทางโดยใช้บัฟเฟอร์แบบหมุนวน (circular buffer) ที่ทางออกแต่ละทาง เพื่อจำกัดช่วงเวลาคงที่ให้สำหรับ guaranteed traffic ในช่วงเวลาหนึ่งวินโดว์ หนึ่งหน่วยของบัฟเฟอร์ จะถูกใช้งานสำหรับหนึ่งแพ็กเกตที่ออกมาจากการกรองของวินโดว์โดยจะเก็บแพ็กเกตนั้นในบัฟเฟอร์ ณ ตำแหน่งที่สำรองไว้ให้การใช้หลักการของวินโดว์ จะช่วยจำกัดขนาดของบัฟเฟอร์ตามจำนวนของแพ็กเกตที่สามารถส่งออกได้ในเส้นทางการเชื่อมโยงขาออกด้วย การจัดการหน่วยความจำชั่วคราวหรือบัฟเฟอร์จะทำการเก็บแพ็กเกตเพื่อรอการส่งออกไปยังโหนดถัดไปประกอบด้วย บัฟเฟอร์สำหรับ ทราฟฟิกที่ต้องการการรับประกันคุณภาพ บัฟเฟอร์สำหรับทราฟฟิกที่มีอัตราเร็วไม่คงที่และมีแพ็กเกตเข้ามามากกว่าจำนวนช่องสัญญาณเวลาที่จองไว้ และบัฟเฟอร์ร่วมซึ่งสำรองไว้สำหรับ ทราฟฟิกที่ไม่ต้องการการรับประกันคุณภาพและกรณีที่บัฟเฟอร์อื่นเต็มก็สามารถใช้งานบัฟเฟอร์ร่วมนี้ได้ซึ่งการพิจารณาเพื่อส่งออกที่พอร์ตใด ๆ จะให้บริการทีละ 1 ช่องสัญญาณเวลาเท่านั้น แล้วจึงเลื่อนไปให้บริการในตำแหน่งช่องสัญญาณเวลาถัดไป การเลื่อนตำแหน่งนี้จะสัมพันธ์กับเวลา ในการเลื่อนของวินโดว์ทิศทางการเลื่อนจะเป็นแบบหมุนวนเช่นเดียวกันเนื่องจากไม่จำเป็นต้องสำรองบัฟเฟอร์ไว้ให้สิ้นเปลืองแบนด์วิดท์ การให้บริการจึงขึ้นอยู่กับการจัดสรรทางเวลาเท่านั้น ยกเว้นบัฟเฟอร์ร่วมซึ่งจะมีขนาดไม่จำกัด ทั้งนี้ขึ้นอยู่กับการร้องขอเพื่อใช้บริการสำหรับ ทราฟฟิกประเภทที่ไม่ต้องการการรับประกันคุณภาพการให้บริการ และต้องทำการสำรองเผื่อกรณีที่บัฟเฟอร์อื่นเต็มด้วย

ส่วนที่เป็นการจัดการโครงสร้างของเฟรมนั้น จะพิจารณาเวลาโดยแบ่งออกเป็นเฟรมและช่องสัญญาณเวลา ซึ่งอาจจะกำหนดให้มีขนาดคงที่หรือไม่คงที่ ตัวอย่างการจัดการเฟรมขนาดคงที่ เช่น Stop-and-Go Queuing และ Quasi-STM ซึ่งจะเป็นการจัดการบนพื้นฐานของเฟรม โดยกำหนดให้เฟรมมีขนาดคงที่ และในหลักการ W-TDM นั้นจะมีช่องสัญญาณเวลาหลายช่องสัญญาณ เพื่อรองรับการให้บริการใน 1 เฟรม และทำการจัดสรรช่องสัญญาณเวลาในเฟรมขนาดคงที่ ซึ่งจะมีผลต่อการทำการซิงโครนัสและทำให้การจัดการง่ายขึ้น แพ็กเกตที่เข้ามาจะถูกเก็บในบัฟเฟอร์ในช่วง 1 หรือ 2 เฟรมเพื่อให้เกิดการซิงโครนัสระหว่างช่องสัญญาณเวลาของเส้นทางเชื่อมโยงขาเข้า (incoming link) และช่องสัญญาณเวลาของเส้นทางเชื่อมโยงขาออกของทุก ๆ ช่องสื่อสาร การเตรียมการรับประกันคุณภาพของการบริการในระบบสื่อสาร B-ISDN นั้นจะต้องมี

รูปแบบการสื่อสารและมีการจัดการช่องสัญญาณเวลาซึ่งแยกการรับประกันคุณภาพการบริการตามประเภทของการสื่อสารในแต่ละทราฟฟิกที่มาจากการเชื่อมโยงต่างๆเช่นการให้บริการการสื่อสารแบบอัตราเร็วคงที่ (Constant Bit Rate, CBR) จะเป็นเหมือนการสื่อสารระบบ TDM เนื่องจากการให้บริการแบบ TDM นั้น มีการสำรองช่องสัญญาณเวลาให้แบบคงที่ และมีการใช้งานช่องสัญญาณเวลาตามที่ได้ทำการจองเอาไว้ และใช้ Time switch ในการจัดแพ็กเกตลงในเฟรม และให้บริการเฉพาะช่องสื่อสารที่ได้รับการสำรองช่องสัญญาณเวลาไว้ให้เท่านั้น วิธีการส่งแบบนี้เป็นผลดีกับการแยกทราฟฟิกอย่างชัดเจน เพราะไม่มีการอนุญาตให้ทราฟฟิกอื่นมาใช้งานช่องสัญญาณเวลาที่ถูกจองไว้แล้ว แต่ข้อเสียคือทำให้การจัดการเป็นไปแบบตายตัวเกิดการสูญเสียแบนด์วิดท์ในแต่ละช่องสื่อสารได้หากมีช่องสัญญาณเวลาว่างและช่องสื่อสารอื่นไม่สามารถมาร่วมใช้ช่องสัญญาณเวลาที่ว่างนั้นได้ จึงทำให้การใช้งานทรัพยากรในระบบไม่เต็มที่ดังนั้นในระบบกึ่งอะซิงโครนัสอย่าง Q-STM และ W-TDM จึงมีการแยกการรับประกันเฉพาะ ทราฟฟิกที่เป็นแบบอัตราเร็วคงที่โดยจองช่องสัญญาณสำหรับทราฟฟิกที่ต้องการการรับประกันคุณภาพการบริการนี้ก่อน แล้วจึงแบ่งช่องสัญญาณเวลาที่เหลือสำหรับให้บริการทราฟฟิกประเภทอื่นๆ ต่อไป

ณ เวลาขณะหนึ่งวินโดว์จะมีขอบเขตบนเฟรมโดยครอบคลุมจำนวนช่องสัญญาณเวลาจำนวนหนึ่งทั้งนี้เพื่อให้ยืดหยุ่นต่อการให้บริการแก่ทราฟฟิกที่สำรองช่องสัญญาณเวลาไว้ให้ โดยการทำงานของวินโดว์จะทำการบริการและจะเลื่อนไปทีละช่องสัญญาณเวลาในทิศทางเดิมเสมอ ซึ่งวิธีการนี้จะคล้ายกับการจัดการแบบ Round Robin แต่มีขอบเขตอยู่ภายในขอบเขตของวินโดว์ นั่นคือแพ็กเกตที่เข้ามาในช่วงเวลาของวินโดว์จะได้รับบริการส่งผ่านไปยังสวิตช์เพื่อส่งออกและการพิจารณาช่องสัญญาณเวลาที่จะสำรองไว้ให้แพ็กเกตในแต่ละช่องสื่อสารนั้นจะต้องอยู่ในขอบเขตของเฟรมที่กำหนด ถ้าไม่พบแพ็กเกต ณ ช่วงเวลาดังกล่าวแพ็กเกตนั้นจะถูกลบทิ้ง ส่วนนี้เป็นข้อเสียของระบบ W-TDM ที่อาจจะทำให้เกิดการสูญหายของแพ็กเกต แต่ทั้งนี้ก็ขึ้นอยู่กับลักษณะพฤติกรรมของทราฟฟิกนั้น ๆ แต่วิธีการนี้ก็เพิ่มความยืดหยุ่นในการให้กับการบริการแบบ TDM และช่วยให้ใช้ช่องสัญญาณเวลาได้เต็มที่ในแกนเวลา ทำให้มีขอบเขตของเวลาหน่วงและทำให้เกิดการแกว่งของเวลาหน่วงน้อยลงด้วย แล้วใช้การจัดการบัฟเฟอร์เพื่อลดความน่าจะเป็นของการสูญหายของแพ็กเกตข้อมูลได้.

บทที่ 3
การพัฒนาอัลกิริทึมการหาเส้นทาง โดยใช้ข้อมูลตำแหน่งช่องสัญญาณเวลา

3.1 นำเสนออัลกอริทึม

งานวิจัยนี้มีวัตถุประสงค์เพื่อสนับสนุนการรับประกันคุณภาพการให้บริการ ตั้งแต่ขั้นตอนการหาเส้นทางในระบบที่มีการจองช่องสัญญาณเวลาตายตัวในเฟรมขนาดคงที่ ซึ่งในระบบสื่อสารแบบนี้การรักษาการซิงโครนัสระหว่างการรับส่งเป็นสิ่งที่สำคัญมาก การที่จะรักษาการซิงโครนัสได้จำเป็นต้องรู้สถานะการใช้งานช่องสัญญาณเวลาเพื่อเลือกเส้นทางที่มีช่องสัญญาณเวลาว่างในตำแหน่งที่จะรักษาการซิงโครนัสได้ตลอดทั้งเส้นทางตั้งแต่โหนดต้นทางถึงโหนดปลายทาง เพราะจะทำให้สามารถจองช่องสัญญาณเวลาได้ในตำแหน่งที่ดีในขั้นตอนการร้องขอเส้นทางเชื่อมโยงและอนุมัติใช้งานเส้นทาง

เนื่องจากสถานะการใช้งานช่องสัญญาณเวลานั้นมีการเปลี่ยนแปลงตลอดเวลาดังนั้นจึงต้องแจ้งข้อมูลสถานะนี้เป็นระยะ ดังนั้นจึงทำการพัฒนาวิธีการเทียบบิตข้อมูลสถานะการใช้งานช่องสัญญาณเวลาลงไปในอัลกอริทึมการหาเส้นทางที่มีการส่งข้อมูลสถานะการใช้งานเส้นทางเชื่อมโยงเป็นระยะ เพื่อให้สามารถส่งสถานะการใช้งานช่องสัญญาณเวลาไปกับแพ็กเกตแจ้งสถานะการใช้งานเส้นทางเชื่อมโยงด้วย จึงได้เลือกใช้อัลกอริทึมการหาเส้นทางแบบสั้นที่สุดและเลือกใช้การหาเส้นทางที่เราท์เตอร์ต้นทางเพราะข้อมูลของทั้งระบบจะอยู่ที่เราเตอร์ต้นทาง ซึ่งจะรู้ข้อมูลโครงสร้างของระบบเครือข่ายภายในเครือข่ายย่อยเดียวกันและสามารถค้นหาที่อยู่ของโหนดปลายทางที่อยู่ในเครือข่ายย่อยอื่นได้โดยผ่านทางเราท์เตอร์หลักของกลุ่มใด ๆ การคำนวณล่วงหน้าที่เราท์เตอร์ต้นทางจะทำให้เลือกเส้นทางที่มีความน่าจะเป็นที่จะจองช่องสัญญาณเวลาที่เหมาะสมได้ และหากเลือกเส้นทางได้เหมาะสมมีทรัพยากรที่เพียงพอและมีตำแหน่งช่องสัญญาณเวลาว่างที่ทำให้เกิดการซิงโครนัสก็จะช่วยลดอัตราการปฏิเสธในกระบวนการร้องขอเพื่อเส้นทางเชื่อมโยงได้ การพิจารณาตำแหน่งซิงโครนัสนั้นต้องพิจารณาจากเฟรมของโหนดส่งและโหนดรับรวมทั้งนำค่าเวลาหน่วงต่าง ๆ มาพิจารณาด้วยเพื่อประมาณค่าตำแหน่งที่แพ็กเกตจะเดินทางมาถึง และตรวจสอบสถานะการใช้งานช่องสัญญาณเวลาได้ว่าตำแหน่งช่องสัญญาณเวลานั้นว่างหรือไม่ การพิจารณาเฟรม การค้นหาเส้นทาง และการเลือกช่องสัญญาณเวลาจะอธิบายรายละเอียดในหัวข้อถัดไป

3.1.1 **การจัดการเฟรมและพิจารณาเวลาหน่วง** (Frame Strategy and Delay Constrain)

ในงานวิจัยนี้ได้เลือกการกำหนดให้เวลาถูกแบ่งออกเป็นเฟรมและกำหนดให้เฟรมมีขนาดคงที่เท่ากับ 125 ไมโครวินาที ซึ่งเป็นมาตรฐานของการสื่อสารแบบใช้เฟรมหลาย ๆ วิธี โดยเฟรมจะถูกแบ่งออกเป็นหลายช่องสัญญาณเวลา ซึ่ง 1 ช่องสัญญาณเวลาจะมีขนาดสำหรับการรับส่ง 1 เซลล์ ATM ซึ่งมีขนาด 53 ไบท์ ดังนั้นเพื่อที่จะรองรับการสื่อสารที่มีอัตราเร็วในการส่งข้อมูลเป็น 155 เมกะบิตต่อวินาที ใน 1 เฟรมจะแบ่งจำนวนช่องสัญญาณเวลาออกเป็น 46 ช่องสัญญาณเวลาโดยคำนวณได้ดังสมการที่ 3.1

$$\text{จำนวนช่องสัญญาณเวลา} = \frac{(155 \times 10^{6}) \times (125 \times 10^{-6})}{53 \times 8} \qquad (3.1)$$

เพื่อให้เป็นไปตามระบบการสื่อสารแบบที่มีการกำหนดช่องสัญญาณเวลาในเฟรมคงที่และเพื่อให้ง่ายต่อการควบคุมจึงกำหนดขนาดเฟรม จำนวนช่องสัญญาณเวลา และขนาดวินโดว์ให้คงที่ ซึ่งในงานวิจัยนี้กำหนดให้ขนาดของวินโดว์เท่ากับ 10 ช่องสัญญาณเวลา ซึ่งมีขนาดประมาณ 1 ใน 4 ของเฟรมทั้งนี้เพื่อให้มีความยืดหยุ่นของช่วงเวลาให้บริการแต่ไม่ให้เพิ่มเวลาหน่วงในกระบวนการจัดลำดับแพ็กเกตเพื่อส่งออกมากเกินไป ส่วนการพิจารณาจำนวนช่องสัญญาณเวลาและการใช้งานช่องสัญญาณเวลานั้นจะใช้ชุดข้อมูลบิตแสดงสถานะการใช้งานโดยกำหนดให้ชุดข้อมูลหนึ่งชุดมีจำนวนบิตเท่ากับจำนวนช่องสัญญาณเวลาในหนึ่งเฟรม ดังนั้นชุดข้อมูลบิตจึงมีจำนวน 46 บิต โดย 1 บิตแทนสถานะการใช้งานหนึ่งช่องสัญญาณเวลา และกำหนดค่าบิตเป็น 1 เพื่อบอกสถานะช่องสัญญาณเวลาว่าง และกำหนดค่าบิตเป็น 0 เพื่อบอกสถานะช่องสัญญาณเวลาถูกจองหรือกำลังให้บริการ ซึ่งชุดข้อมูลบิตนี้จะถูกส่งไปยังโหนดต่าง ๆ โดยส่งไปพร้อมกับแพ็กเกตแจ้งสถานะการใช้งานเส้นทางเชื่อมโยง ดังนั้นชุดข้อมูลบิตแสดงสถานะการใช้งานช่องสัญญาณเวลานี้จึงมีการแจ้งเพื่อปรับปรุงข้อมูลเป็นระยะตามคาบเวลาของการส่งแพ็กเกตแจ้งสถานะการใช้งานเส้นทางเชื่อมโยง ข้อมูลนี้จะถูกบันทึกและนำไปใช้ในกระบวนการคำนวณหาเส้นทางและการสำรองช่องสัญญาณเวลาในแต่ละสวิตชิ่งโหนดต่อไป

การส่งข้อมูลระหว่างสวิตชิ่งโหนดใด ๆ จะเกิดเวลาหน่วงค่าหนึ่งซึ่งจะต่างกันไปตามระยะทางระหว่างโหนดส่งกับโหนดรับและขึ้นอยู่กับเวลาในการจัดการแถวคอยและกระบวนการจัดการแพ็กเกตของแต่ละโหนด ซึ่งเวลาหน่วงนี้จะมีผลต่อการซิงโครนัสของการรับส่งแพ็กเกตและอาจทำให้แพ็กเกตมาไม่ตรงกับตำแหน่งช่องสัญญาณเวลาที่จองไว้ได้ เวลาหน่วงต่าง ๆ ที่มีผลกระทบต่อการรับส่งแพ็กเกตข้อมูลมีดังนี้

1. τ เวลาในการส่งผ่านแพ็กเกตข้อมูล 1 แพ็กเกต มีหน่วยเป็นวินาที และกำหนดให้เป็นคาบเวลาในการให้บริการ 1 ช่องสัญญาณเวลา

$$\tau = \frac{l}{R} \tag{3.2}$$

l เป็นขนาดของแพ็กเกต (packet length) ซึ่งกำหนดให้มีขนาดเท่ากับเซล ATM มีขนาดเท่ากับ 53 x 8 = 424 บิต

R เป็นอัตราเร็วในการส่งผ่านแพ็กเกต (transmission rate)

R = 155 Mbps ดังนั้น τ = 2.735 μs

2. Δt เวลาหน่วงเนื่องจากการส่งแพ็กเกตข้อมูลผ่านสื่อ (propagation delay) เป็นค่าคงที่หาได้จากอัตราส่วนของระยะทางกับอัตราเร็วในการส่งข้อมูลผ่านสื่อ ดังนั้นเวลาหน่วงเนื่องจากการส่งแพ็กเกตข้อมูลผ่านสื่อระหว่างโหนด i และโหนด j ใด ๆ มีค่าเป็น

$$\Delta t_{ij} = \frac{d_{ij}}{v} \tag{3.3}$$

d_{ij} เป็นระยะทางระหว่างโหนด i และโหนด j มีหน่วยเป็นเมตร

v เป็นอัตราเร็วในการส่งแพ็กเกตข้อมูลผ่านสื่อ หากเป็นสายนำสัญญาณแบบไฟเบอร์ อัตราเร็วนี้จะมีค่าเท่ากับความเร็วแสง (Light velocity) ซึ่งมีความเร็วอยู่ที่ 3 x 10^8 เมตรต่อวินาที

3. Δq เป็นเวลาหน่วงเนื่องจากการรอในแถวคอย หากทุกแพ็กเกตเข้ามาถึง ณ ตำแหน่งที่จองไว้ก็จะได้รับการบริการทันที ซึ่งจะทำให้ไม่มีแพ็กเกตค้างในแถวคอยดังนั้น Δq = 0 หากมีแพ็กเกตค้างในแถวคอยจะต้องนำจำนวนแพ็คเกตที่ค้างอยู่ทั้งหมดมาหาค่าเวลาหน่วงเนื่องจากการรอในแถวคอย ดังนี้

$$\Delta q = c \times \tau \tag{3.4}$$

c เป็นจำนวนแพ็กเกตที่รออยู่ในแถวคอยขณะนั้น

4. Δp เวลาที่ใช้ในการเข้าถึงส่วนหัวของแพ็กเกตแล้วทำการเปรียบเทียบค่า VCI ในแพ็กเกตกับค่า VCI ในตารางวงจรเสมือน ทั้งนี้เพื่อเปลี่ยนเป็นค่า VCI ที่จะถูกนำไปใช้ในกระบวนการเดียวกันนี้ในสวิตชิ่งโหนดถัดไป แล้วส่งแพ็กเกตข้อมูลไปเก็บยังบัฟเฟอร์ของพอร์ตขาออกเพื่อรอส่งออกไปยังโหนดถัดไป โดยกำหนดให้เวลาในการจัดการแพ็กเกตข้อมูลหนึ่งแพ็กเกตตั้งแต่ได้รับแพ็กเกตเข้ามา และตรวจสอบข้อมูลในตารางวงจรเสมือนจนถึงการส่งแพ็กเกตข้อมูลออกไปยังบัฟเฟอร์ที่ทางออกดังสมการที่ 3.5

$$\overline{\Delta p} = a \cdot t_a \qquad (3.5)$$

t_a เป็นเวลาในการเข้าถึงข้อมูลแบบเรียงตามลำดับ(Sequential Search)ของข้อมูล 1 ตำแหน่งในตารางวงจรเสมือน

$a = \frac{w}{2}$ เป็นความน่าจะเป็นเฉลี่ยของจำนวนครั้งในการค้นหาข้อมูลในตารางวงจรเสมือน โดยที่ w เป็นจำนวนช่องสัญญาณเวลาในวินโดว์

5. Δm เป็นเวลาในการรับส่วนข้อมูลของแพ็กเกต ซึ่งจะดำเนินไปในขณะเดียวกันกับกระบวนการจัดการส่วนหัวของแพ็กเกตในขั้นตอนการจัดลำดับแพ็กเกตเพื่อส่งออก (Δp) โดย Δm มีค่าเป็น

$$\Delta m = \frac{b}{R} \qquad (3.6)$$

b เป็นจำนวนบิตข้อมูล (payload) มีขนาดเท่ากับ 48 x 8 = 384 บิต

Δm และ Δp เป็นกระบวนการที่เกิดขึ้นพร้อมกัน ดังนั้นจึงต้องพิจารณาเลือกเวลาหน่วงที่มีค่ามากกว่า

$$\Delta s = \max(\Delta m, \Delta p) \qquad (3.7)$$

ดังนั้นเวลาหน่วงที่เกิดขึ้นในกระบวนการรับส่งข้อมูลระหว่างโหนด i และโหนด j จะเป็นดังนี้

$$\Delta T_{ij} = \Delta q + \Delta t_{ij} + \Delta s \qquad (3.8)$$

ΔT_{ij} เป็นเวลาหน่วงที่จะบังคับให้เกิดการซิงโครนัสในการรับส่งระหว่างโหนด i และโหนด j ได้โดยนำเวลานี้มาคำนวณเพื่อหาตำแหน่งการจองช่องสัญญาณเวลาให้สอดคล้องกับตำแหน่งช่องสัญญาณเวลาที่แพ็กเกตจากโหนด i จะมาถึงยังโหนด j ซึ่งจะทำให้สามารถเข้าถึงข้อมูลในตารางการจัดลำดับแพ็กเกตในตำแหน่งปัจจุบันและได้รับการบริการทันที

กระบวนการรับส่งข้อมูลจะเริ่มจากการกำหนดช่องสัญญาณเวลาเริ่มต้นในเฟรมของทุกพอร์ตในทุกโหนดเป็นช่องสัญญาณเวลาที่ 0 ณ เวลา t_0 และกำหนดให้วินโดว์เริ่มพิจารณาที่ตำแหน่งนี้เช่นเดียวกัน ดังนั้นการเลื่อนของวินโดว์จึงสัมพันธ์กัน และการจองช่องสัญญาณเวลาจึงต้องคำนวณเวลาหน่วงที่แพ็กเกตจะไปถึงยังโหนดใด ๆ จากนั้นจึงสามารถนำเวลาหน่วงนี้มาคำนวณหาตำแหน่งของวินโดว์ ณ ขณะนั้นได้ว่าเลื่อนไปอยู่ที่ช่องสัญญาณเวลาใด ซึ่งจะสามารถหาตำแหน่งเยื้องในการซิงโครนัสระหว่างเฟรมของโหนดส่ง i และโหนดรับ j ได้ดังสมการที่ 3.9 ตำแหน่งเยื้องนี้เป็นตัวแปรควบคุมที่สำคัญที่บอกถึงตำแหน่งซิงโครนัสระหว่างโหนด i และโหนด j ใด ๆ ซึ่งพิจารณาจากเวลาหน่วงในการส่งผ่านระบบและกระบวนการต่าง ๆ ภายในสวิตช์ ตำแหน่งเยื้องนี้จะถูกนำไปใช้ในการพิจารณาเพื่อเทียบบิตข้อมูลในขั้นตอนการหาเส้นทางและการหาตำแหน่งในการจองช่องสัญญาณเวลา

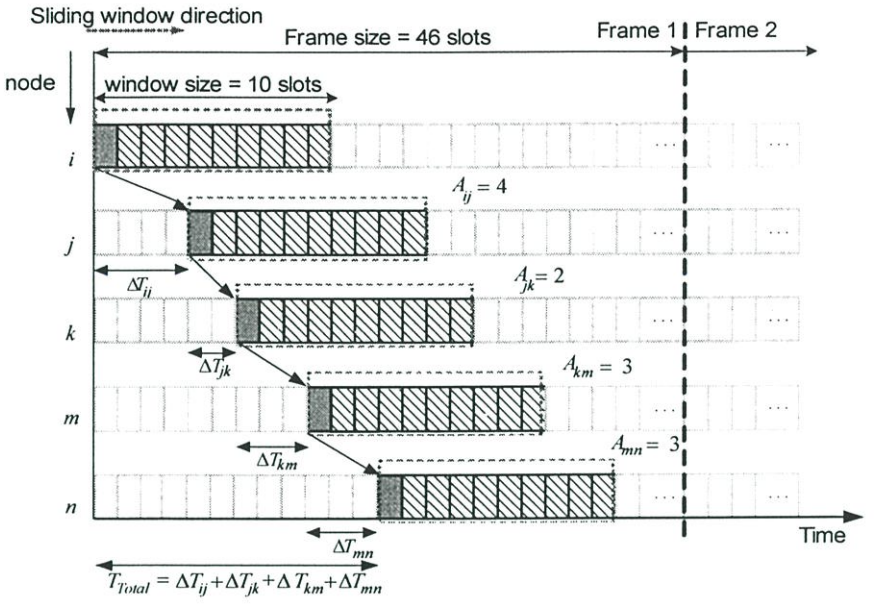
$$A_{ij} = \left(\frac{\Delta T_{ij}}{\tau} \right) \bmod f \qquad (3.9)$$

A_{ij} เป็นตำแหน่งเยื้องที่จะทำให้เกิดการซิงโครนัสระหว่างโหนด i และโหนด j

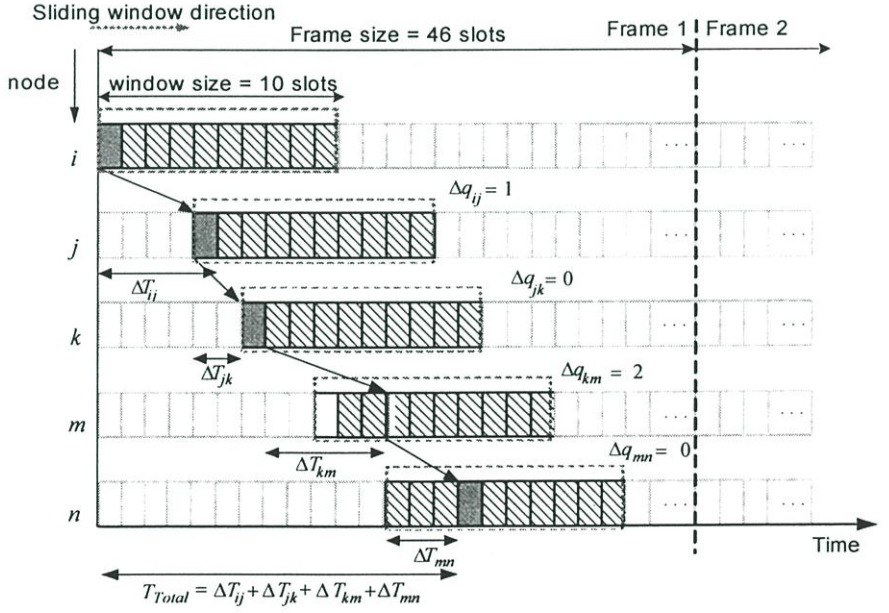
ΔT_{ij} เป็นเวลาหน่วงของการรับส่งข้อมูลระหว่างโหนด i และโหนด j

f เป็นจำนวนช่องสัญญาณเวลาทั้งหมดใน 1 เฟรม

จากรูปที่ 3.1 มีการจองช่องสัญญาณเวลาที่ตำแหน่งซ้ายสุดของวินโดว์ซึ่งเป็นตำแหน่งที่ซิงโครนัสในการรับส่งระหว่างสองโหนด จะเห็นว่าช่องสัญญาณเวลาที่ 0 ของโหนด i ซิงโครนัสกับช่องสัญญาณเวลาที่ 4 ของโหนด j ดังนั้นตำแหน่งเยื้องเท่ากับ 4 และช่องสัญญาณเวลาที่ 4 ของโหนด j ซิงโครนัสกับช่องสัญญาณเวลาที่ 6 ของโหนด k ตำแหน่งเยื้องเท่ากับ 2 ส่วนการซิงโครนัสระหว่างโหนด k กับโหนด m และระหว่างโหนด m กับโหนด n นั้นมีตำแหน่งเยื้องเท่ากับ 3 ซึ่งตำแหน่งเยื้องที่ซิงโครนัสนี้เป็นตำแหน่งแรกที่เหมาะแก่การจองช่องสัญญาณเวลาเพราะหากแพ็กเกตเข้ามาถึง ณ เวลานี้จะได้รับการบริการทันที แต่ในระบบการใช้งานเครือข่ายจริงตำแหน่งดังกล่าวอาจถูกจองโดยทราฟฟิกอื่นที่เข้ามาก่อนได้ จึงเลือกใช้หลักการ W-TDM มารับประกันคุณภาพการบริการเพื่อเพิ่มความยืดหยุ่นของตำแหน่งช่องสัญญาณที่สามารถจองได้



รูปที่ 3.1 การพิจารณาเวลาหน่วงควบคุมและตำแหน่งเยื้องที่ทำให้เกิดการซิงโครนัสในเส้นทางจากโหนด i ไปยังโหนด n



รูปที่ 3.2 การพิจารณาเวลาหน่วงเมื่อไม่สามารถจองในตำแหน่งซ้ายสุดของวินโดว์

การจองช่องสัญญาณเวลาจะพิจารณาตำแหน่งช่องสัญญาณเวลาที่ซิงโครนัสซึ่งเป็นตำแหน่งแรกในวินโดว์และพิจารณาได้ตั้งแต่ตำแหน่งแรกนี้ไปจนถึงขอบเขตของวินโดว์ ซึ่งการจองนี้จะมีผลกระทบต่อเวลาหน่วงด้วยดังรูปที่ 3.2 การเลือกช่องสัญญาณเวลาที่ต้องการจองอาจเลือกตำแหน่งที่ซิงโครนัสโดยตรง หรือทำการสุ่มเพื่อเลือกตำแหน่งใดตำแหน่งหนึ่งในขอบเขตวินโดว์

การสุ่มเพื่อเลือกตำแหน่งภายในวินโดว์นี้จะลดความน่าจะเป็นของความต้องการจองช่องสัญญาณเวลาในตำแหน่งที่ตรงกันของทราฟฟิกต่าง ๆ ที่เข้ามาขอเชื่อมโยงเส้นทางได้ แต่จะทำให้ Δq มากกว่าการเลือกตำแหน่งที่ซิงโครนัส หากแพ็กเกตเข้ามาตรงตำแหน่งช่องสัญญาณเวลาที่ซิงโครนัส แพ็กเกตจะได้รับการบริการโดยทันทีและมีเวลาหน่วงที่น้อยที่สุดในการส่งผ่านเส้นทางจากโหนด i ถึงโหนด n ซึ่งมีค่าเท่ากับ ΔT_{in} แต่หากเป็นการจองในตำแหน่งที่ไม่ซิงโครนัสกันและทำให้เวลาหน่วง Δp มากกว่า Δm แล้ว เวลาหน่วงทั้งระบบจะเพิ่มขึ้นเป็น $\Delta T_{in} + \Delta q$ การให้บริการนั้นหากแพ็กเกตมาถึงหลังเวลาให้บริการหรือมาหลังจากที่วินโดว์เลื่อนไปแล้ว แพ็กเกตนั้นจะถูกลบทิ้งซึ่งการสูญหายของแพ็กเกตในลักษณะนี้สามารถแก้ไขได้โดยใช้บัฟเฟอร์ร่วมแต่แพ็กเกตก็ต้องรอรับบริการในรอบเวลาของวินโดว์ถัดไปซึ่งจะวนกลับมาถึง ณ ตำแหน่งนี้อีกครั้ง

3.1.2 การพัฒนาวิธีการหาเส้นทางโดยใช้ข้อมูลตำแหน่งช่องสัญญาณเวลา

ประกอบด้วย

1. การเก็บข้อมูลสถานะการใช้งานช่องสัญญาณเวลา
2. การส่งข้อมูลสถานะการใช้งานช่องสัญญาณเวลาไปกับแพ็กเกตแจ้งสถานะการใช้งานเส้นทางเชื่อมโยง
3. การปรับค่าข้อมูลและค่าภาระการใช้ลิงค์หลังจากที่ได้รับข้อมูลจากแพ็กเกตแจ้งสถานะการใช้งานเส้นทางเชื่อมโยง
4. การพิจารณาเพื่อเลือกเส้นทางที่เหมาะสม

<u>การเก็บข้อมูลสถานะการใช้งานช่องสัญญาณเวลา</u>

สถานะการใช้งานช่องสัญญาณเวลาที่สนใจคือสถานะการใช้งานช่องสัญญาณเวลาในเฟรมของพอร์ตขาออก เนื่องจากพอร์ตขาออกเป็นพอร์ตที่จะส่งแพ็กเกตไปยังโหนดถัดไป ดังนั้นจึงต้องแน่ใจว่าช่องสัญญาณเวลาในเฟรมของพอร์ตนั้นมีตำแหน่งที่ว่างซึ่งซิงโครนัสระหว่างการส่งออกจากโหนดปัจจุบันและการรับของโหนดข้างเคียง และตำแหน่งดังกล่าวนี้จะสัมพันธ์กับการสำรองตำแหน่งในตารางวงจรเสมือน โดยวิธีการจัดลำดับแพ็กเกตเพื่อส่งออกนั้นใช้หลักการ W-TDM ซึ่งสวิตช์ในหลักการ W-TDM นั้นจะมีตารางวงจรเสมือนแยกตามพอร์ตขาเข้า และใช้การพิจารณาตำแหน่งในเฟรมของพอร์ตขาออกใด ๆ มาเป็นตำแหน่งที่สัมพันธ์กับการจองในตารางวงจรเสมือน เช่น แพ็กเกตเข้ามาทางพอร์ตขาเข้าที่ 1 แต่ต้องการออกทางพอร์ตขาออกที่ 3 จะต้องพิจารณาข้อมูลการใช้งานช่องสัญญาณเวลาในเฟรมของพอร์ตขาออกที่ 3 เพื่อหาตำแหน่งว่างที่เหมาะสมกับการจองแล้วจึงนำค่าตำแหน่งนั้นมาเป็นตำแหน่งที่เก็บค่า VCI ในตารางวงจรเสมือน เมื่อแพ็กเกตข้อมูลเข้ามาทางพอร์ตขาเข้าที่ 1 ณ เวลาที่จองไว้ จะถูกส่งไปทำกระบวนการจัดลำดับ

แพ็กเกตเพื่อส่งออกโดยการเข้าถึงตารางวงจรเสมือน ณ ตำแหน่งปัจจุบัน ซึ่งหากแพ็กเกตมีการรับส่งอย่างซิงโครนัสจะหมายความว่าตำแหน่งนี้จะเป็นตำแหน่งที่แพ็กเกตจะได้รับการบริการทันที หรือมีการแปลข้อมูลส่วนหัวของแพ็กเกต และถูกส่งออกไปยังบัฟเฟอร์ที่เฟรมของพอร์ตขาออกที่ 3 ในตำแหน่งที่สัมพันธ์กับตำแหน่งดังกล่าว และจะถูกส่งออกไปได้ทันที

สถานะการใช้งานช่องสัญญาณเวลา 1 ช่องสัญญาณจะถูกแทนด้วยหนึ่งบิต โดยค่าบิตเท่ากับ 0 บอกสถานะช่องสัญญาณเวลาไม่ว่าง และค่าบิตเท่ากับ 1 บอกสถานะช่องสัญญาณเวลาว่าง โดยค่าสถานะการใช้งานช่องสัญญาณเวลาจะถูกกำหนดไว้ครั้งแรกเป็นบิต 1 ทั้งเฟรม เนื่องจากเป็นสถานะว่างทั้งเฟรม ทุกครั้งที่มีการเปลี่ยนแปลงสถานะ เช่น มีการอนุมัติการจองช่องสัญญาณเวลา หรือมีการแจ้งยกเลิกการใช้งานช่องสัญญาณเวลา จะต้องมีการเก็บข้อมูลสถานะการใช้งานช่องสัญญาณเวลาใหม่เสมอ ดังนั้นเราท์เตอร์และสวิตชิ่งโหนดทุกตัวจะต้องปรับข้อมูลสถานะการใช้งานช่องสัญญาณเวลาทุกครั้งที่มีการเปลี่ยนแปลง แล้วส่งข้อมูลที่เป็นปัจจุบันไปพร้อมกับแพ็กเกตแจ้งสถานะการใช้งานเส้นทางเชื่อมโยงทุกช่องสัญญาณเวลาหรือมีการเปลี่ยนแปลงมากเกินระดับที่กำหนด

<u>การส่งข้อมูลสถานะการใช้งานช่องสัญญาณเวลา</u>

วิธีการ SUBM นั้นเป็นการคำนวณล่วงหน้าในขั้นตอนการหาเส้นทาง ซึ่งจะต้องรู้ข้อมูลการใช้งานช่องสัญญาณเวลาของทุกโหนดในเส้นทาง ดังนั้นโหนดที่ทำหน้าที่ตัดสินใจเลือกเส้นทางจะต้องรู้ข้อมูลสถานะการใช้งานช่องสัญญาณเวลาของทุกโหนดในระบบ ซึ่งอัลกอริทึมการหาเส้นทางแบบพิจารณาสถานะการใช้งานเส้นทางเชื่อมโยงนั้นจะมีการส่งแพ็กเกตแจ้งสถานะการใช้งานเส้นทางเชื่อมโยงเป็นระยะ ดังนั้นจึงเลือกพัฒนาวิธีการ SUBM เพิ่มเติมลงในอัลกอริทึมการหาเส้นทางแบบที่มีการพิจารณาสถานะการใช้งานเส้นทางเชื่อมโยง (Link state algorithm) ซึ่งเหมาะกว่าวิธีการหาเส้นทางแบบเวกเตอร์ระยะทางที่รู้เพียงข้อมูลเส้นทางที่คำนวณแล้วของโหนดข้างเคียงเท่านั้น แพ็กเกตแจ้งสถานะการใช้งานเส้นทางเชื่อมโยงนี้จะถูกส่งครั้งแรกเมื่อเริ่มมีการใช้งานระบบและจะถูกเซตคาบเวลาในการส่งไว้ด้วยเพื่อให้แพ็กเกตถูกส่งออกไปเป็นระยะ ๆ เราท์เตอร์และ สวิตชิ่งโหนดทุกตัวจะทำการสุ่มทางเวลาเพื่อหาเวลาเริ่มต้นในการส่งแพ็กเกตแจ้งสถานะการใช้งานเส้นทางเชื่อมโยง แต่คาบเวลาจะถูกเซตให้มีค่าเท่ากันเพื่อให้การปรับค่าข้อมูลในเราท์เตอร์และสวิตชิ่งโหนดอื่น ๆ เกิดขึ้นไม่พร้อมกัน กรณีที่มีการเปลี่ยนแปลงโครงสร้างในเครือข่ายเช่นมีขนาดของระบบขยายขึ้น มีการเชื่อมโยงของอุปกรณ์เพิ่มขึ้นในระบบ จะต้องมีการส่งข้อมูลเพื่อไปปรับค่าข้อมูลเสมอ เมื่อมีการร้องขอใช้งานเส้นทางเชื่อมโยงและจองช่องสัญญาณเวลาในสวิตชิ่งโหนดแต่ละตัวก็จะมีการเก็บค่าสถานะใหม่ เพื่อรอส่งออกไปกับแพ็กเกตแจ้งสถานะการใช้งานเส้นทางเชื่อมโยงในครั้งถัดไป

การปรับค่าข้อมูลและค่าภาระการใช้ลิงค์

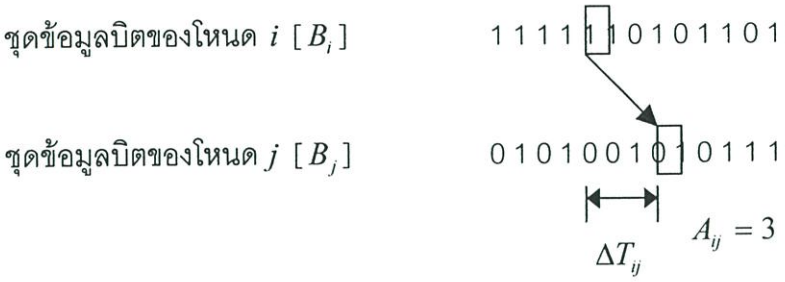
หลังจากที่ได้รับข้อมูลจากแพ็กเกตแจ้งสถานะการใช้งานเส้นทางเชื่อมโยง สวิตชิ่งโหนดจะทำการเข้าถึงข้อมูลในแพ็กเกตแจ้งสถานะการใช้งานเส้นทางเชื่อมโยงและเก็บค่าตัวแปรที่มากับแพ็กเกต ซึ่งที่สวิตชิ่งโหนดนี้จะมีตารางเก็บข้อมูลสถานะและค่าตัวแปรต่าง ๆ โดยแยกเก็บตามโหนดที่ส่งแพ็กเกตมา ส่วนข้อมูลสถานะการใช้งานช่องสัญญาณเวลานั้นจะมีการเก็บข้อมูลแยกตามพอร์ตอีกที ทุกครั้งที่ได้รับแพ็กเกตเข้ามาจะทำการตรวจหาหมายเลขของโหนดต้นทางที่ส่งแพ็กเกตมาและตรวจสอบหมายเลขลำดับเฉพาะของแพ็กเกต ซึ่งหากเป็นหมายเลขลำดับเดิมที่เคยได้รับแล้ว ก็จะทำการลบแพ็กเกตนั้นทิ้ง แต่หากเป็นหมายเลขลำดับที่ใหม่กว่าก็จะทำการเก็บข้อมูล ดังนั้นจึงมีการปรับปรุงข้อมูลใหม่ทุกครั้งที่ได้รับแพ็กเกตแจ้งสถานะการใช้งานเส้นทางเชื่อมโยงใหม่ การพิจารณาข้อมูลสถานะการใช้งานช่องสัญญาณเวลาจะมีอยู่ 2 ส่วน คือการนำข้อมูลมาคำนวณหาภาระลิงค์และการเปรียบเทียบ SUBM เพื่อพิจารณาหาช่องสัญญาณเวลาว่างที่ซิงโครนัสระหว่างโหนดต่าง ๆ ในเส้นทางหรือช่องสัญญาณเวลาที่สามารถให้บริการได้ภายใต้ขอบเขตของวินโดว์ (มี Δq) การเปรียบเทียบบิตนั้นจะทำในกระบวนการหาเส้นทางที่เราเตอร์ซึ่งจะอธิบายในหัวข้อต่อไป ส่วนการรับแพ็กเกตแจ้งสถานะการใช้งานเส้นทางเชื่อมโยงจะทำเพียงเก็บข้อมูลแยกตามโหนดที่ส่งแพ็กเกตมาและปรับค่าภาระลิงค์

กรณีที่เป็นการนำข้อมูลมาคำนวณหาค่าภาระลิงค์ หากแพ็กเกตแจ้งสถานะการใช้งานเส้นทางเชื่อมโยงที่รับเข้ามาเป็นของโหนดข้างเคียงสวิตชิ่งโหนดปัจจุบัน จะมีการคำนวณค่าภาระลิงค์เพื่อปรับค่าภาระลิงค์ให้กับเส้นทางเชื่อมโยงที่เชื่อมระหว่างโหนดข้างเคียงกับโหนดโหนดปัจจุบัน ซึ่งค่าภาระลิงค์ในวิธีการนี้จะประมาณการได้จากอัตราส่วนของเวลาหน่วงเนื่องจากการส่งแพ็กเกตข้อมูลผ่านสื่อต่อจำนวนช่องสัญญาณเวลาว่างของเส้นทางเชื่อมโยงนั้นดังสมการ 3.10

$$\text{cost} = \frac{\Delta t_{ij}}{s_{e_{ij}}} \qquad (3.10)$$

Δt_{ij} เป็นเวลาหน่วงในการส่งแพ็กเกตข้อมูลผ่านสื่อระหว่างโหนด i และโหนด j

$s_{e_{ij}}$ เป็นจำนวนช่องสัญญาณเวลาว่างในตำแหน่งที่ซิงโครนัสกันระหว่างโหนด i และโหนด j ซึ่งทำได้โดยการเปรียบเทียบข้อมูลสถานะการใช้งานช่องสัญญาณเวลาที่ซิงโครนัสของพอร์ตที่เชื่อมต่ออยู่กับเส้นทางเชื่อมโยงระหว่างโหนด i และโหนด j เมื่อ ΔT_{ij} เป็นเวลาหน่วงรวมของการรับส่งแพ็กเกตข้อมูลระหว่างโหนด i และโหนด j ที่ทำให้เกิดการซิงโครนัสกันระหว่างการรับส่งแพ็กเกต และ A_{ij} เป็นตำแหน่งเยื้องที่ทำให้เกิดการซิงโครนัสระหว่างโหนด i และโหนด j สามารถอธิบายการเทียบบิตได้ดังนี้



กำหนดให้เป็น C_{ij} เป็นชุดบิตข้อมูลซึ่งมีการเรียงบิตสถานะการใช้งานช่องสัญญาณเวลาของทางออกในโหนด j [B_j] ใหม่ โดยเลื่อนวนบิต (circular shift left) ของโหนด j ไปทางซ้าย A_{ij} ตำแหน่งเพื่อให้สัมพันธ์กับข้อมูลสถานะของโหนด i ซึ่งจะได้

1 0 0 1 0 1 0 1 1 1 0 1 0 1 0 0 . . . 1 0 1 0 1 0

ผลการเทียบบิต ((B_i) AND (C_{ij})) 1 0 0 1 0 1 0 1 0 1 0 0 0 0 0 0 . . . . . 0 1 0

เมื่อนับจำนวนผลลัพธ์ของการเทียบบิตที่เป็น 1 ก็จะได้จำนวนช่องสัญญาณเวลาว่างที่ซิงโครนัสกันในเฟรมที่รับส่งผ่านเส้นทางเชื่อมโยงระหว่าง โหนด i และโหนด j และทำการคำนวณค่าภาระลิงค์ดังสมการ 3.10 ในลักษณะเดียวกันนี้ทุกเส้นทางเชื่อมโยงที่เชื่อมระหว่างโหนดข้างเคียงกับโหนดปัจจุบันและทำการปรับค่าภาระลิงค์ให้กับเส้นทางเชื่อมโยงเหล่านั้นก่อนส่งแพ็กเกตกระจายออกไปทางพอร์ตอื่นที่ไม่ใช่พอร์ตที่รับแพ็กเกตเข้ามา

<u>การพิจารณาเพื่อเลือกเส้นทางที่เหมาะสม</u>

การหาเส้นทางขั้นแรกจะใช้วิธีการหาเส้นทางที่สั้นที่สุด นั่นคือเป็นเส้นทางที่มีค่าภาระการใช้ลิงค์รวมทั้งเส้นทางน้อยที่สุด การพิจารณาจะเริ่มจากโหนดต้นทางโหนดหนึ่งโหนดใด ๆ แล้วเลือกเส้นทางที่มีค่าภาระการใช้ลิงค์รวมต่ำสุด แต่ในการเก็บเส้นทางจะเก็บเส้นทางที่เป็นไปได้ไว้จำนวนหนึ่ง โดยเรียงลำดับตามค่าภาระการใช้ลิงค์รวมของเส้นทางจากน้อยไปหามาก และหากเส้นทางมีค่าภาระการใช้ลิงค์รวมเท่ากัน ก็จะพิจารณาเรียงลำดับตามจำนวน Hop ในเส้นทางต่อไป เส้นทางและค่าภาระการใช้ลิงค์ของเส้นทางเหล่านั้นจะถูกนำมาใช้พิจารณากรณีที่ทำการเทียบบิตแล้วไม่มีช่องสัญญาณเวลาที่ซิงโครนัส โดยนำเส้นทางที่มีค่าภาระการใช้ลิงค์รวมน้อยที่สุดและมีจำนวน Hop น้อยที่สุดมาพิจารณาเปรียบเทียบบิตข้อมูลเพื่อหาตำแหน่งซิงโครนัสระหว่างเฟรมของโหนดในเส้นทาง เพื่อหาตำแหน่งที่ควรจะทำการจองใช้งานช่องสัญญาณเวลา ซึ่งจะได้อธิบายการจองต่อไป หากเส้นทางแรกที่เป็นเส้นทางที่สั้นที่สุดไม่มีตำแหน่งว่างที่ซิงโครนัสกันเลยก็จะนำเอาเส้นทางถัดไปมาพิจารณาต่อไป ดังนั้นการเทียบบิตข้อมูลเพื่อหาตำแหน่งซิงโครนัสระหว่างเฟ

รมของโหนดในเส้นทางจะทำได้โดยการนำข้อมูลสถานะการใช้งานช่องสัญญาณเวลาของพอร์ตที่เชื่อมอยู่กับเส้นทางเชื่อมโยงระหว่างโหนดในเส้นทางมาทำการเปรียบเทียบบิตข้อมูลโดยการ AND เช่น เส้นทางที่ได้จาก โหนด i ถึงโหนด n มีทิศทางผ่านโหนดต่าง ๆ ดังนี้

$$i \text{ -> } j \text{ -> } k \text{ -> } m \text{ -> } n$$

ชุดข้อมูลบิต (46 บิต ต่อ 1 ชุดข้อมูล)

	0	1	2	3	4	5	6	7	8	9	···	45
B_i	1	1	1	0	1	0	1	1	0	1	···	0
C_{ij}	0	1	0	1	1	0	1	1	0	1	···	0
C_{ik}	1	1	1	1	0	1	0	1	1	0	···	1
C_{im}	1	0	1	0	0	1	0	1	0	0	···	0
C_{in}	1	0	1	0	0	1	0	1	0	1	···	0
			c			b		a				

รูปที่ 3.3 การเทียบบิตโดยการ AND ชุดบิตข้อมูลในเส้นทางจากโหนด i ไปยังโหนด n

เมื่อโหนด i เป็นเราท์เตอร์ต้นทางและหาเส้นทางไปยังโหนด n ซึ่งป็นเราท์เตอร์ปลายทาง ข้อมูลสถานะการใช้งานช่องสัญญาณเวลาที่นำมาพิจารณาจะเริ่มจากบิตข้อมูลของโหนด i ในพอร์ตที่เชื่อมอยู่กับเส้นทางเชื่อมโยงระหว่างโหนด i และโหนด j และใช้ ΔT_{ij} คำนวณหาตำแหน่งเยื้อง A_{ij} ที่ทำให้เกิดการซิงโครนัสระหว่างโหนด i และโหนด j ซึ่งจะเรียงชุดบิตข้อมูลใหม่ได้เป็น C_{ij} ข้อมูลถัดไปจะเป็นชุดบิตข้อมูลของโหนด j ในพอร์ตที่เชื่อมอยู่กับเส้นทางเชื่อมโยงระหว่างโหนด j และโหนด k ซึ่งตำแหน่งเยื้องจะพิจารณาตั้งแต่เวลาหน่วงที่เริ่มส่งจากโหนด i นั่นคือใช้ ΔT_{ik} มาหาตำแหน่งเยื้อง A_{ik} ซึ่งเรียงชุดบิตข้อมูลใหม่ได้เป็น C_{ik} และบิตข้อมูลของโหนด k ในพอร์ตที่เชื่อมอยู่กับเส้นทางเชื่อมโยงระหว่างโหนด k และโหนด m ใช้ ΔT_{im} มาหาตำแหน่งเยื้อง A_{im} ซึ่งเรียงชุดบิตข้อมูลใหม่ได้เป็น C_{im} และสุดท้ายจะเป็นบิตข้อมูลของโหนด m ในพอร์ตที่เชื่อมอยู่กับเส้นทางเชื่อมโยงระหว่างโหนด m และโหนด n ใช้ ΔT_{in} มาหาตำแหน่งเยื้อง A_{in} ซึ่งเรียงชุดบิตข้อมูลใหม่ได้เป็น C_{in} และเมื่อเลื่อนบิตซิงโครนัสให้ตรงกันแล้วจะสามารถเทียบข้อมูลบิตสถานะการใช้งานช่องสัญญาณเวลาทั้งเส้นทางได้ดังรูปที่ 3.3 นำบิตข้อมูลมาเรียงและทำการเลื่อนบิตเพื่อให้ตำแหน่งซิงโครนัสตรงกันทุกโหนดแล้วจึงทำการเทียบบิตข้อมูลเพื่อหา

ตำแหน่งว่างที่ซิงโครนัสในเส้นทางซึ่งอาจจะมีได้มากกว่าหนึ่งตำแหน่ง การจองตำแหน่งช่องสัญญาณเวลาที่ดีที่สุดคือการจองแบบ a ที่ช่องสัญญาณเวลาที่ 7 เนื่องจากมีตำแหน่งช่องสัญญาณเวลาว่างที่ซิงโครนัสกันทั้งเส้นทางนั่นเอง ซึ่งตำแหน่งดังกล่าวนี้ค่าเวลาหน่วงรวมจะน้อยที่สุด ซึ่งเวลาหน่วงในเส้นทางจะพิจารณาได้ดังนี้

1. การจองช่องสัญญาณเวลาในแบบ a ช่องสัญญาณเวลาที่ใช้ได้คือช่องสัญญาณเวลาที่ 7 เวลาหน่วงเท่ากับ

$$\Delta T_{Total} = \Delta T_{in} = \Delta T_{ij} + \Delta T_{jk} + \Delta T_{km} + \Delta T_{mn} \qquad (3.11)$$

2. การจองช่องสัญญาณเวลาในแบบ b และ c ทั้ง 2 กรณี มีการเยื้องตำแหน่งการจองไป 1 บิต ดังนั้น

$$\Delta T_{Total} = \Delta T_{in} + \Delta q = \Delta T_{in} + \tau \qquad (3.12)$$

ในงานวิจัยนี้กำหนดการสำรองช่องสัญญาณเวลาสำหรับทุกทราฟฟิกมีจำนวนเท่า ๆ กัน คือเท่ากับ 1 ช่องสัญญาณเวลา ทั้งนี้เพื่อให้ง่ายในการพิจารณาในเรื่องของการหาเส้นทาง ดังเช่นตัวอย่างในรูปที่ 3.3 การจองช่องสัญญาณเวลาที่เหมาะที่สุดคือการจองในแบบ a ซึ่งเมื่อเลือกช่องสัญญาณเวลาได้แล้ว เราท์เตอร์ i ก็จะทำการเก็บเส้นทางนี้เป็นเส้นทางที่เหมาะสมที่สุด ที่จะไปยังโหนด n และเก็บตำแหน่งที่สามารถจองได้ไว้ในตารางเส้นทางด้วย โดยตำแหน่งที่เก็บ จะเป็นตำแหน่งที่จะจองที่เราท์เตอร์ต้นทางนี้เอง ตัวอย่างในรูปที่ 3.3 จะเก็บค่าตำแหน่งช่องสัญญาณเวลาที่ 7 ไว้ หากมีแพ็กเกตร้องขอการเชื่อมโยงเส้นทางไปยังโหนด n ในกระบวนการจองช่องสัญญาณเวลาจะมาตรวจสอบตำแหน่งในตารางเส้นทางและทำการจองให้ทราฟฟิกนี้ ที่ตำแหน่งดังกล่าวโดยเก็บค่าหมายเลขพอร์ตและค่า VCI ที่จะไปยังโหนด j ไว้ในตารางวงจรเสมือน

เมื่อทำการจองเสร็จแล้วจะต้องทำการเทียบบิตอีกครั้งซึ่งการเทียบบิตครั้งนี้เส้นทางที่ดีที่สุดจากโหนด i ไปยังโหนด n อาจต่างไปจากเดิมเนื่องจากข้อมูลบิตภายในโหนดมีการเปลี่ยนแปลงและในการเทียบบิตใหม่นี้จะต้องปรับค่าบิตสถานะการใช้งานเส้นทางเชื่อมโยงของโหนดในเส้นทางซึ่งสามารถคาดการณ์ได้ว่าโหนดเหล่านั้นจะถูกจองในตำแหน่งใดบ้างจากการเทียบบิตในการเลือกเส้นทางก่อนหน้านี้ ซึ่งการปรับค่าข้อมูลบิตของโหนดเหล่านี้จะทำให้ข้อมูลที่นำมาเปรียบเทียบใกล้เคียงข้อมูลล่าสุดซึ่งใช้แทนข้อมูลจริงที่ยังส่งมาไม่ถึง ดังนั้นเส้นทางที่ไปยังโหนด n อาจจะเป็นเส้นทางเดิมหรือเป็นเส้นทางใหม่ก็ได้ขึ้นอยู่กับข้อมูลช่องสัญญาณเวลา และการปรับค่าตัวแปรทราฟฟิกและค่าภาระการใช้ลิงค์จากข้อมูลที่ได้รับจากแพ็กเกตแจ้งสถานะการใช้งานเส้นทาง

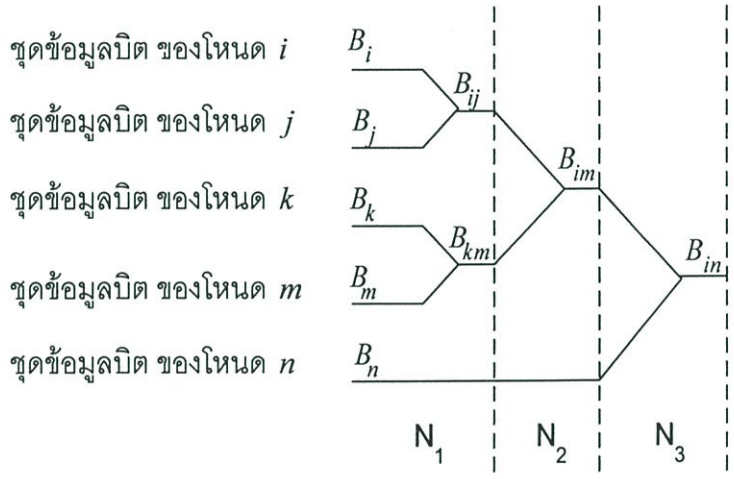
เชื่อมโยงหากเป็นเส้นทางเดิมจะทำการเลือกสุ่มช่องสัญญาณเวลาที่ซิงโครนัสในตำแหน่งอื่นมาเก็บไว้ในตารางเส้นทาง หากเป็นเส้นทางใหม่ก็เก็บเส้นทางใหม่และตำแหน่งที่สามารถจองได้ไว้ในตารางเส้นทางและก่อนที่จะส่งแพ็กเกตร้องขอเส้นทางเชื่อมโยงออกไปจะต้องทำการคำนวณตำแหน่งที่สามารถจองได้ที่โหนดถัดไปโดยนำค่าเวลาหน่วงของการส่งจากโหนด i ไปยังโหนด j มาพิจารณาหาตำแหน่งเยื้องแล้วนำค่าตำแหน่งดังกล่าวมาบวกกับตำแหน่งที่จองในโหนดปัจจุบัน เมื่อคำนวณตำแหน่งที่น่าจะทำการจองที่โหนดถัดไปได้แล้วก็จะเก็บค่าตำแหน่ง ดังกล่าวลงไปในแพ็กเกตด้วย ซึ่งที่สวิตชิ่งโหนดถัดไปจะนำค่าตำแหน่งนี้ไปใช้ในการจอง หากพบว่าตำแหน่งดังกล่าวนี้ไม่ว่างก็จะทำการคำนวณตำแหน่งที่น่าจะจองในโหนดถัดไปอีก ทำเช่นนี้จนกระทั่งถึงเราท์เตอร์ปลายทาง

กรณีที่ต้องการจองแบนด์วิดท์สองช่องสัญญาณจะต้องทำการจองช่องสัญญาณเวลาในแบบ b และ c ทั้งสองช่องสัญญาณจึงจะได้เส้นทางที่ทำให้เกิดการซิงโครนัสระหว่างเฟรมของสวิตชิ่งโหนดและไม่เกิดการแกว่งของเวลาหน่วงด้วย การแกว่งของเวลาหน่วงจะเกิดขึ้นก็ต่อเมื่อมีการจองช่องสัญญาณเวลามากกว่า 1 ช่องสัญญาณและมีการจองในลักษณะที่แตกต่างกัน หากทำการจองช่องสัญญาณเวลาในแบบที่ 1 และแบบที่ 2 สำหรับทราฟฟิกเดียวกันจะทำให้เกิดการแกว่งของเวลาหน่วงได้เพราะเวลาหน่วงของการมาถึงของแพ็กเกตทั้งสองครั้งไม่เท่ากัน ดังนั้นจึงเกิดการแกว่งของเวลาหน่วงขึ้นได้ ฉะนั้นการจองกรณีที่ต้องการช่องสัญญาณเวลามากกว่า 1 ช่องสัญญาณจะต้องพยายามหาเส้นทางที่มีลักษณะการจองที่เหมือนกัน จึงจะทำให้ไม่เกิดการแกว่งของเวลาหน่วง การพิจารณาการจองและการกระจายการจองจำเป็นต้องมีกระบวนการที่ซับซ้อนกว่านี้ ซึ่งสามารถนำไปทำการวิจัยได้ต่อไป และเพื่อลดความยุ่งยากที่จะเกิดขึ้นจากการจองหลายช่องสัญญาณ จึงได้กำหนดให้ 1 ทราฟฟิกสามารถใช้งานได้ 1 ช่องสัญญาณเวลา การนำข้อมูลบิตแสดงสถานะการใช้งานช่องสัญญาณเวลามาคำนวณล่วงหน้าในกระบวนการหาเส้นทางจะทำให้ได้เส้นทางที่มีช่องสัญญาณเวลาว่างและมีแบนด์วิดท์เพียงพอ มีการซิงโครนัสระหว่างเฟรมของสวิตชิ่งโหนดในเส้นทาง ทำให้แพ็กเกตมีการรับส่งตรงเวลาจึงได้รับการรับประกันคุณภาพในการส่งผ่านข้อมูลและลดเวลาหน่วงจากการปฏิเสธในกระบวนการสร้างเส้นทางเชื่อมโยง ดังนั้นวิธีการหาเส้นทางแบบนี้จึงเหมาะกับระบบสื่อสารที่มีการกำหนดช่องสัญญาณเวลาตายตัวในเฟรมแบบคงที่ กรณีที่ทำการเปรียบเทียบบิตข้อมูลแล้วไม่มีตำแหน่งว่างที่ซิงโครนัสกันในขั้นตอนแรกจะยังไม่เลือกใช้เส้นทางนั้นเพราะอาจเกิดการปฏิเสธการร้องขอเส้นทางเชื่อมโยง เนื่องจากไม่มีตำแหน่งที่สามารถจองได้ และหากจองในตำแหน่งที่ไม่เหมาะสมก็อาจจะทำให้เกิดการสูญหายของแพ็กเกตได้ ดังนั้นเมื่อทำการเปรียบเทียบบิตแล้วไม่มีตำแหน่งว่างจำเป็นจะต้องคำนวณในลักษณะที่มีการเลื่อนบิตแล้ว AND อีกครั้ง เพราะในหลักการของ W-TDM นั้นสามารถจองตำแหน่งที่เยื้องจากตำแหน่งที่ซิงโครนัสได้หากยังอยู่ในช่วงของวินโดว์ และการเลื่อนบิตแล้ว

AND นี้บางครั้งจะพบว่ามีช่องสัญญาณเวลาว่างเพิ่มขึ้นด้วย แต่การเลื่อนบิตจะทำให้ค่าเวลาหน่วงเปลี่ยนไป ดังนั้นทุกครั้งที่เลื่อนบิตเพื่อเปรียบเทียบต้องตรวจสอบเวลาหน่วงกับค่าสูงสุดของเวลาหน่วงที่กำหนด (Maximum End-to-End delay) ซึ่งค่านี้จะเป็นไปตามความต้องการในการรับประกันคุณภาพการให้บริการแก่ทราฟฟิก ขั้นตอนการเทียบบิตข้อมูลสามารถอธิบายได้ดังนี้

1. หาตำแหน่งเยื้องที่ซิงโครนัสระหว่างโหนดที่อยู่ติดกันตามสมการที่ 3.9

2. ทำการ AND ชุดข้อมูลบิตเป็นคู่ดังรูปที่ 3.4 โดยที่ B_i แทนชุดข้อมูลบิตของโหนด i และ B_{ij} เป็นผลการ AND ชุดบิตข้อมูลของโหนด i และโหนด j ซึ่งต้องทำการเลื่อนบิตซิงโครนัสให้ตรงกันก่อนทำการ AND ทุกครั้งโดย N คือ ลำดับขั้นของการ AND การพิจารณาชุดข้อมูลบิตเป็นคู่นี้จะเป็นประโยชน์ในกรณีที่ผลการ AND ไม่มีบิตที่ซิงโครนัสจะทำการเลื่อนตำแหน่งบิตข้อมูลเพื่อหาตำแหน่งที่ซิงโครนัสให้ได้ก่อนที่จะทำการเทียบบิตกับโหนดคู่ถัดไป



รูปที่ 3.4 วิธีการเทียบชุดข้อมูลบิตโดยทำการ AND เป็นคู่

การเทียบบิตในรูปที่ 3.4 มีลำดับการทำงานดังนี้

1. ชุดข้อมูล B_i ทำการ AND กับชุดข้อมูล B_j จะได้ชุดข้อมูล B_{ij} หากชุดข้อมูล B_{ij} ที่ได้เป็น 0 ทั้งหมดจะต้องทำการเลื่อนบิตข้อมูล B_j แล้วทำการ AND อีกครั้ง การเลื่อนบิตจะทำทีละบิตและเลื่อนไปทางซ้ายในลักษณะหมุนวนแล้วตรวจสอบผลการ AND จนกว่าจะมีผลลัพธ์ไม่เท่ากับ 0 ซึ่งแสดงว่ามีตำแหน่งว่างที่ซิงโครนัสให้สามารถจองได้ ทั้งนี้ต้องพิจารณาเวลาหน่วงทุกครั้งที่มีการเลื่อนบิตข้อมูล โดยคำนวณและเก็บค่าเวลาหน่วงที่เพิ่มขึ้นเพื่อตรวจสอบเวลาหน่วงทั้งระบบไม่ให้เกินค่าสูงสุดของเวลาหน่วงที่กำหนด

2. เมื่อทำการเลื่อนชุดข้อมูล B_j ไปแล้ว ชุดข้อมูลผลลัพธ์ของโหนดถัดไปทุกโหนดจะต้องทำการเลื่อนไปทางซ้ายตามจำนวนบิตที่เลื่อนไปด้วยเพื่อรักษาตำแหน่งการซิงโครนัส ดังนั้นเมื่อทำ

การ AND ชุดบิตข้อมูล B_i และ B_j ได้เป็นชุดข้อมูล B_{ij} แล้ว ก่อนที่จะทำการ AND กับชุดข้อมูล B_{km} จะต้องทำการเลื่อนบิตของชุดข้อมูล B_{km} ไปเท่ากับจำนวนบิตของชุดข้อมูล B_j ที่เลื่อนไปบวกกับตำแหน่งเยื้อง A_{jk} ด้วย แล้วจึงทำการ AND และเมื่อ AND กันได้เป็นชุดข้อมูล B_{im} แล้วเป็น 0 ทั้งหมดก็สามารถเลื่อนบิตข้อมูล B_{km} แล้ว AND ใหม่อีกครั้ง โดยไม่จำเป็นต้องทำการ AND ชุดข้อมูล B_k และ B_m ใหม่

3. ลำดับสุดท้ายนำข้อมูล B_{im} มา AND กับ B_n จะต้องทำการเลื่อนบิตของชุดข้อมูล B_n ตามจำนวนบิตที่ได้ทำการเลื่อนทั้งหมดรวมกับตำแหน่งเยื้อง A_{mn} ดังนั้นระหว่างที่ทำกระบวนการเทียบบิตนี้ต้องมีการเก็บค่าเวลาหน่วงและบิตที่เลื่อนทุกครั้งแล้วบวกค่าเวลาหน่วงเมื่อมีการเลื่อนบิตเพื่อจะได้ตรวจสอบเวลาหน่วงและตรวจสอบการซิงโครนัสได้ตลอดทั้งเส้นทาง

3.2 การออกแบบและการพัฒนาอัลกอริทึมลงในอุปกรณ์เครือข่ายจำลอง

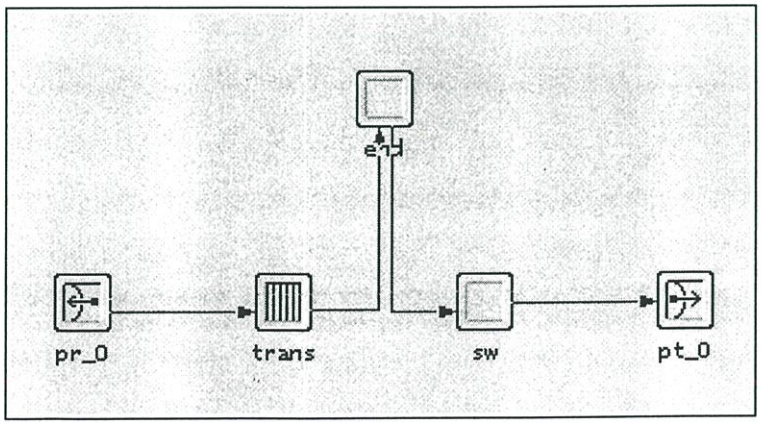
จากอัลกอริทึมที่กล่าวมาในหัวข้อ 3.1 ต้องนำมาทำการออกแบบและสร้างระบบเครือข่ายจำลองเพื่อทำการตรวจสอบถึงประสิทธิภาพของวิธีการดังกล่าว การออกแบบระบบเครือข่ายจำลองเพื่อใช้ในงานวิจัยครั้งนี้จะประกอบด้วย การออกแบบอุปกรณ์เครือข่ายจำลอง การออกแบบโครงสร้างระบบเครือข่ายจำลองและออกแบบการทดลอง ซึ่งในบทนี้จะอธิบายการออกแบบอุปกรณ์เครือข่ายจำลองต่าง ๆ ส่วนโครงสร้างระบบเครือข่ายจำลองและวิธีการทดลองนั้นจะอธิบายในบทที่ 4 อุปกรณ์จำลองที่ต้องใช้ในระบบเครือข่ายจำลองประกอบด้วย 3 อุปกรณ์หลัก ๆ ได้แก่ อุปกรณ์สื่อสารปลายทางซึ่งทำหน้าที่เป็นทั้งโหนดส่งและโหนดรับ อุปกรณ์เราท์เตอร์จำลองซึ่งจะทำงานในชั้นควบคุมเครือข่าย ทำการพิจารณาแพ็กเกตควบคุม คำนวณหาเส้นทาง และการจัดลำดับแพ็กเกตเพื่อส่งออกไปตามเส้นทาง และอุปกรณ์สวิตช์จำลองซึ่งจะมีฟังก์ชันการทำงานคล้ายกับอุปกรณ์เราท์เตอร์ เพียงแต่ไม่สามารถหาเส้นทางได้เนื่องจากระบุให้ระบบเครือข่ายนี้ใช้การหาเส้นทางที่เราท์เตอร์ต้นทางเท่านั้น ดังนั้นสวิตช์จึงมีหน้าที่เพียงสลับแพ็กเกตไปตามเส้นทางที่กำหนดไว้เท่านั้น ส่วนอุปกรณ์สื่อสารปลายทางนั้นโหนดต้นทางจะมีความต้องการสร้างและส่งข้อมูลไปยังโหนดปลายทาง แล้วจะทำการแจ้งความต้องการในการร้องขอเพื่อเชื่อมโยงเส้นทาง (call request) โดยสร้างแพ็กเกตร้องขอการเชื่อมโยงเส้นทางส่งไปยังเราท์เตอร์ต้นทางซึ่งเชื่อมต่ออยู่กับโหนดต้นทางโดยตรงหรือเป็นเราท์เตอร์หลักที่ควบคุมการรับส่งข้อมูลในเครือข่ายย่อยของโหนดต้นทางนั้น ซึ่งที่เราท์เตอร์จะมีข้อมูลเส้นทางไปยังโหนดปลายทางต่าง ๆ ที่เหมาะสมที่สุดอยู่ในตารางเส้นทาง (Routing Table) ดังนั้นเมื่อมีแพ็กเกตร้องขอการเชื่อมโยงเส้นทางเข้ามาที่เราท์เตอร์แล้ว เราท์เตอร์จะทำการเข้าถึงข้อมูลในแพ็กเกตเพื่อหาหมายเลขโหนดปลายทางที่แพ็กเกตต้องการส่งไป แล้วหาเส้นทางที่จะไปยังโหนดปลายทางนั้นในตารางเส้นทาง เมื่อได้เส้นทางที่เหมาะสมแล้วก็จะทำการระบุเส้นทางนั้นลงไปในแพ็กเกตด้วยก่อนที่จะส่งออกไปยังสวิตช์ถัดไป

ถัดไป ซึ่งกระบวนการโดยละเอียดจะกล่าวถึงต่อไป ส่วนสวิตชิ่งโหนดนั้นจะทำหน้าที่เป็นเหมือนอุปกรณ์สวิตช์ที่รับและส่งแพ็กเก็ตออกไปทางพอร์ตที่สัมพันธ์กับเส้นทางที่จะไปยังโหนดปลายทาง มีการจองช่องสัญญาณเวลา การจัดลำดับแพ็กเกตเพื่อส่งออก และการเก็บแพ็กเกตข้อมูลไว้ที่บัฟเฟอร์ขาออกเพื่อรอส่งในเวลาให้บริการเพื่อให้เกิดการซิงโครนัสระหว่างแต่ละโหนดในเส้นทาง เมื่อแพ็กเกตร้องขอการเชื่อมโยงส่งไปถึงยังโหนดปลายทางแล้วโหนดปลายทางจะส่งแพ็กเกตตอบรับกลับมาตามเส้นทางจนถึงโหนดต้นทาง เมื่อโหนดต้นทางได้รับการตอบรับก็จะส่งแพ็กเกตข้อมูลออกมา ซึ่งในงานวิจัยนี้กำนดให้มีการจองในเฟรมตายตัวแบบคงที่ และกำหนดให้ทุกทราฟฟิกใช้งาน 1 ช่องสัญญาณเวลา ดังนั้นทราฟฟิกที่มีอัตราเร็วในการส่งที่ต่างกันต้องมีกระบวนการเก็บข้อมูลเพื่อรอส่งในตำแหน่งที่จองไว้ด้วย เมื่อถึงตำแหน่งช่องสัญญาณเวลาที่จองไว้แพ็กเกตข้อมูลจึงจะถูกส่งออกไปยังสวิตชิ่งโหนด และแพ็กเกตข้อมูลนี้จะถูกสวิตช์และส่งไปตามเส้นทางจนถึงโหนดปลายทางต่อไป

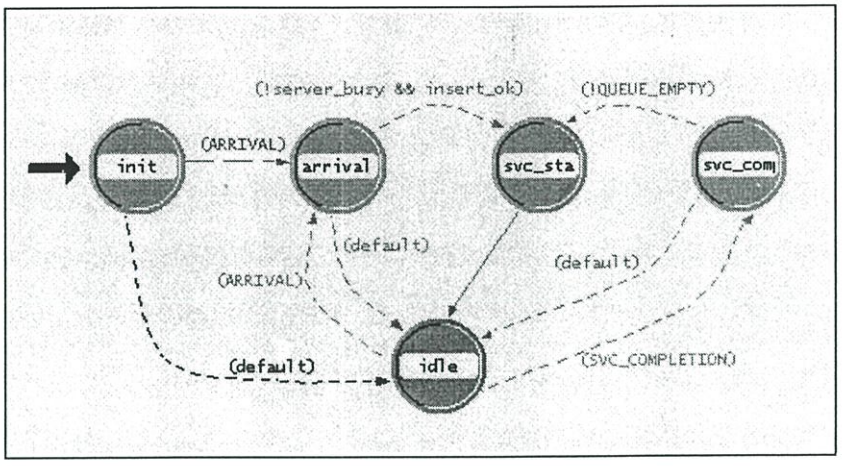
จากรูปแบบการทำงานดังที่กล่าวมาจะนำมาพิจารณารายละเอียดและโครงสร้างของอุปกรณ์ต่าง ๆ โดยละเอียด โดยทำการออกแบบและจำลองบนโปรแกรมจำลองระบบเครือข่าย Opnet เวอร์ชัน 6.0 รายละเอียดโครงสร้างและการทำงานของอุปกรณ์ต่าง ๆ เป็นดังนี้

3.2.1 อุปกรณ์สื่อสารปลายทางจำลอง

โหนดที่ทำหน้าที่เป็นอุปกรณ์สื่อสารปลายทางจะทำงานได้สองอย่างเป็นทั้งโหนดต้นทางและโหนดปลายทางได้ในเวลาเดียวกัน โครงสร้างของโหนดจะประกอบด้วยพอร์ตขาเข้า (pr_0) พอร์ตขาออก (pt_0) และมอดูลต่าง ๆ ดังรูปที่ 3.5 ซึ่งจะอธิบายแต่ละมอดูลย่อยต่าง ๆ ต่อไป



รูปที่ 3.5 แบบจำลองภายในอุปกรณ์สื่อสารปลายทาง (End node model)



รูปที่ 3.6 แผนผังสถานะภายในมอดูล trans (state diagram of trans module)

3.2.1.1 **มอดูล** trans

ที่อุปกรณ์สื่อสารปลายทางนี้ การรับแพ็กเกตจะรับเข้ามาทางพอร์ตขาเข้าซึ่งเชื่อมต่อกับมอดูล trans โครงสร้างการทำงานในมอดูลนี้เป็นดังรูปที่ 3.6 ทำหน้าที่จัดการแพ็กเกตเพื่อรอส่งไปยังมอดูลอื่น ๆ แพ็กเกตทุกประเภทที่รับเข้ามาจะต้องถูกเก็บและรออยู่ในแถวคอยซึ่งเป็นเหมือนบัฟเฟอร์ขาเข้า แล้วจึงส่งแพ็กเกตต่อไปยังมอดูล end ต่อไป จำนวนแพ็กเกตที่รอในแถวคอย (p) หาได้จากค่าเฉลี่ยของอัตราส่วนของอัตราเร็วการส่งข้อมูลเฉลี่ยและอัตราเร็วในการส่งผ่านสายสื่อสาร (R) ซึ่งอัตราเร็วการส่งข้อมูลเฉลี่ยคำนวณได้จากผลคูณของอัตราเร็วของการเข้ามาของแพ็กเกต (λ) มีหน่วยเป็นแ พ็กเกตต่อวินาทีกับความยาวเฉลี่ยของแพ็กเกต (L) ซึ่งมีหน่วยเป็น บิตต่อแพ็กเกต

$$p = \frac{\lambda L}{R} \tag{3.13}$$

และสามารถหาเวลารอคอยเฉลี่ย (Average waiting time, $\bar{u}$) ได้จากอัตราส่วนของจำนวนแพ็กเกตในแถวคอยต่ออัตราเร็วในการให้บริการ (Service rate, r)

$$\bar{u} = \frac{p}{r} \tag{3.14}$$

ซึ่งขนาดของแพ็กเกตจะขึ้นอยู่กับประเภทของแพ็กเกตที่โหนดต้นทางหรือโหนดปลายทางสร้างและส่งออกมา และเนื่องจากความยาวของแพ็กเกตที่เข้ามาไม่เท่ากันดังนั้นจึงต้องหาความ

ยาวเฉลี่ยของข้อมูล และนำมาคูณกับอัตราเร็วเฉลี่ยของการมาถึงของแพ็กเกตเพื่อหาอัตราการมาถึงเฉลี่ยซึ่งมีหน่วยเป็นบิตต่อวินาที จากนั้นจึงนำไปหารด้วยอัตราเร็วของการส่งผ่านสายสื่อสารซึ่งมีหน่วยเป็นบิตต่อวินาทีเช่นเดียวกัน แล้วนำค่าที่ได้มาคำนวณหาค่าเฉลี่ยของเวลาในแถวคอยซึ่งในโปรแกรมเวลารอคอยในแถวคอยนี้จะถูกนำมาเซตค่าให้กับเวลาอินเตอร์รัพท์ (interrupt time) เพื่อนำแพ็กเกตออกจากแถวคอย

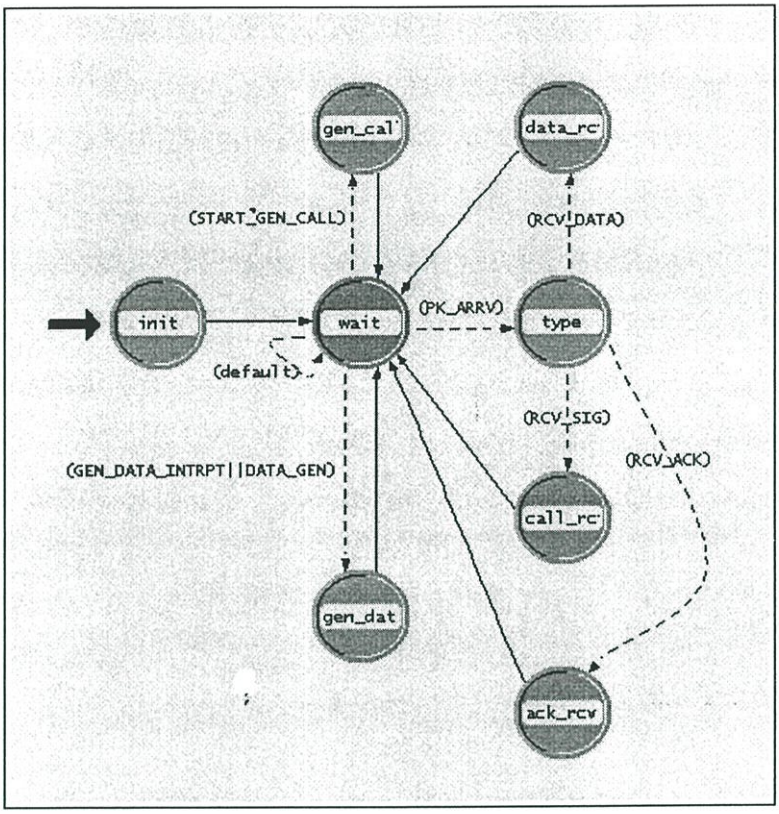
จากแผนผังสถานะสามารถอธิบายได้ โดยเริ่มที่สถานะเริ่มต้น (init state) ซึ่งจะทำการเซตค่าเริ่มต้นของตัวแปรต่าง ๆ แล้วจากนั้นจะไปรอรับเหตุการณ์ต่าง ๆ ที่อาจจะเกิดขึ้น(อินเตอร์รัพท์) ที่สถานะว่าง (idle state) หรือหากมีแพ็กเกตเข้ามาก็จะไปทำงานที่สถานะรับแพ็กเกต (arrival state) ซึ่งจะตรวจสอบแพ็กเกตและเก็บข้อมูลต่าง ๆ ของแพ็กเกตที่สถานะนี้ จากนั้นจึงตรวจสอบการให้บริการว่าขณะนั้นมีการบริการส่งแพ็กเกตอยู่หรือไม่ หากกำลังให้การบริการส่งแพ็กเกตอยู่หรือว่าหากมีแพ็กเกตค้างอยู่ในแถวคอยจะต้องนำแพ็กเกตที่เข้ามาใหม่นี้ไปเก็บและรอในแถวคอยเพื่อรอการให้บริการซึ่งจะเป็นไปตามลำดับการมาถึง ที่สถานะ svc_start จะทำการคำนวณเวลารอคอยและตั้งเวลาในการอินเตอร์รัพท์เพื่อนำแพ็กเกตออกจากแถวคอยและเมื่อถึงเวลาในการอินเตอร์รัพท์จะย้ายไปทำงานที่สถานะ svc_compl มีการนำแพ็กเกตออกจากแถวคอยและส่งออกไปยังมอดูล end จะเห็นว่าทุกสถานะจะเซตค่า default ของการเปลี่ยนสถานะให้เปลี่ยนเป็นสถานะว่าง เนื่องจากหากทำงานในสถานะนั้นเสร็จแล้วต้องมารอเหตุการณ์ที่จะอินเตอร์รัพท์เพื่อไปทำงานยังสถานะอื่นต่อไป

3.2.1.2. **มอดูล** end

มอดูล end จะเป็นกระบวนการทำงานของโหนดสื่อสารปลายทางมีโครงสร้างการทำงานของสถานะต่าง ๆ ดังรูปที่ 3.7 ซึ่งจะอธิบายการทำงานที่สถานะต่าง ๆ ได้ดังนี้

<u>สถานะเริ่มต้น (init state)</u>

ที่สถานะนี้จะทำการสุ่มค่า (random) ทางเวลาในการสร้างแพ็กเกตร้องขอการเชื่อมโยงเส้นทางและค่าเวลานี้จะถูกนำไปเซ็ตให้กับเวลาของการอินเตอร์รัพท์เพื่อสร้างแพ็กเกต การสุ่มค่าเวลานี้จะทำให้เสมือนว่าเป็นความต้องการของแอพลิเคชั่น ซึ่งจะมีการร้องขอเพื่อส่งข้อมูลอย่างไม่แน่นอนและเพื่อแทนความหลากหลายของแอพลิเคชั่นจึงได้กำหนดให้มีการกระจายตัวของทราฟฟิกด้วยฟังก์ชันการกระจาย (Probability Density function, PDF) โดยแบ่งการกระจายออกเป็น แบบอัตราเร็วคงที่ (constant traffic) กับแบบเอกซ์โปเนนเชียล (exponential traffic) การเลือกการกระจายจะใช้การสุ่มเช่นเดียวกัน ดังนั้นทราฟฟิกที่ส่งออกมาจึงไม่ได้มีรูปแบบและลำดับการเกิดทราฟฟิกที่แน่นอน ซึ่งในงานวิจัยนี้จะสนใจทราฟฟิกแบบที่มีอัตราเร็วคงที่ขณะเดียวกันก็ใช้ฟังก์ชันเอกซ์โปเนเชียลสำหรับทราฟฟิกแบบที่มีอัตราเร็วไม่คงที่อื่นๆ

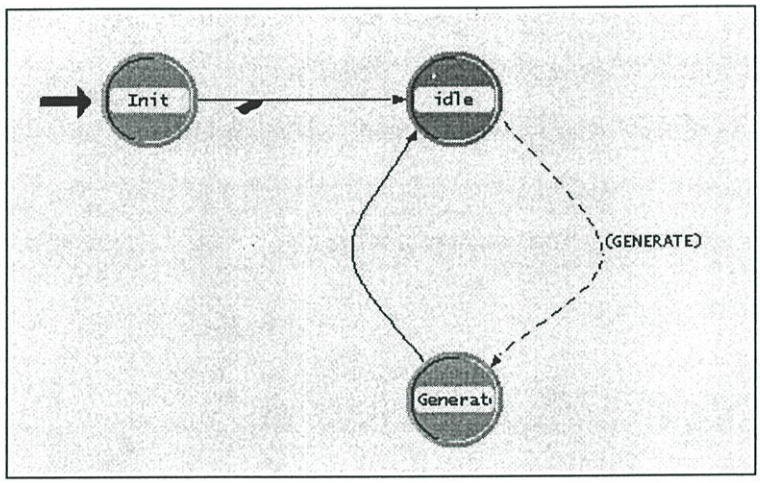


รูปที่ 3.7 แผนผังสถานะของมอดูล end (state diagram of end module)

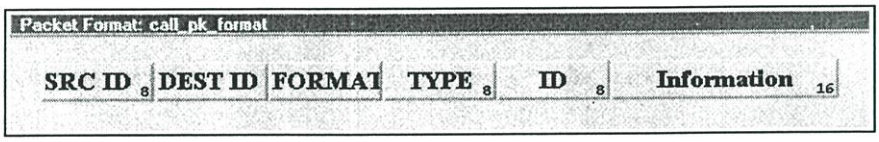
ซึ่งที่สถานะเริ่มต้นนี้ จะทำการสุ่มเพื่อเลือกรูปแบบการกระจายให้กับแต่ละทราฟฟิกที่ต้องการส่งข้อมูล นอกจากนี้ยังมีการสุ่มคาบเวลา (inter-arrival time) ของการสร้างและส่งแพ็กเกตข้อมูลเพื่อให้อัตราเร็วที่ต้องการในการส่งข้อมูลไม่เท่ากัน และสุ่มเลือกเวลาในการหยุดสร้างแพ็กเกต ซึ่งจะแทนความต้องการในการส่งข้อมูลของแต่ละทราฟฟิกให้มีขนาดข้อมูลที่ต่างกัน และเพื่อให้ทราฟฟิกมีการกระจายทิศทางการส่งในระบบเครือข่าย จึงต้องมีการสุ่มค่าหมายเลข (Identified number, ID) ของโหนดที่เป็นอุปกรณ์ปลายทาง ซึ่งหมายเลขนี้แทนแอดเดรสของอุปกรณ์สื่อสารปลายทางในระบบ และหมายเลขที่สุ่มได้นี้จะนำไปเซตค่าแอดเดรสปลายทางให้กับแต่ละทราฟฟิก รายละเอียดต่าง ๆ ของทราฟฟิกที่กล่าวมาข้างต้นนี้จะถูกเก็บลงในตารางข้อมูลการสร้างแพ็กเกตร้องขอการเชื่อมโยงเส้นทาง ส่วนเวลาที่สุ่มเพื่อเป็นระยะห่างของการสร้างแพ็กเกตร้องขอการเชื่อมโยงเส้นทางจะถูกนำไปเก็บเป็นคาบเวลาของการอินเตอร์รัพท์เพื่อสร้างแพ็กเกตร้องขอการเชื่อมโยงเส้นทาง จากนั้นโปรแกรมจะรอเหตุการณ์และเวลาอินเตอร์รัพท์ที่สถานะรอคอย (wait state) เมื่อมีการอินเตอร์รัพท์ให้สร้างแพ็กเกตร้องขอเส้นทางเชื่อมโยง จะย้ายการทำงานไปยังสถานะการสร้างแพ็กเกตร้องขอเส้นทางเชื่อมโยง (gen_call state) หรือหากมีแพ็กเกตเข้ามาก็จะย้ายไปทำงานที่สถานะแยกแพ็กเกต (type state)

สถานะการสร้างแพ็กเกตร้องขอเส้นทางเชื่อมโยง (gen_call state)

เมื่อมีการอินเตอร์รัพท์มาที่สถานะนี้จะมีการตรวจสอบลำดับของข้อมูลทราฟฟิกที่ต้องการสร้างแพ็กเกตร้องขอเพื่อเชื่อมโยงเส้นทาง จากนั้นจะนำข้อมูลไปเก็บในตารางย่อยและส่งค่าตัวชี้ (pointer) ที่ชี้ตารางย่อยนี้ไปพร้อมกับการเรียกกระบวนการทำงานย่อย (invoke child process) ในการสร้างแพ็กเกตร้องขอเพื่อเชื่อมโยงเส้นทาง กระบวนการทำงานย่อยจะใช้ค่าตัวชี้ที่ถูกเซตให้ไปนำข้อมูลในตารางมาสร้างแพ็กเกตออกมาแล้วนำมาใส่ค่าข้อมูลตัวแปรทราฟฟิกลงในฟิลด์ (field) ต่าง ๆ ของแพ็กเกต แผนผังสถานะของกระบวนการย่อยในการสร้างแพ็กเกตร้องขอการเชื่อมโยงแสดงดังรูปที่ 3.8 ซึ่งจะอธิบายการทำงานของกระบวนย่อยในการสร้างแพ็กเกตร้องขอเพื่อเชื่อมโยงเส้นทางได้ดังนี้



รูปที่ 3.8 แผนผังสถานะของกระบวนย่อยในการสร้างแพ็กเกตร้องขอเพื่อเชื่อมโยงเส้นทาง



รูปที่ 3.9 โครงสร้างของแพ็กเกตร้องขอเพื่อเชื่อมโยงเส้นทาง

● สถานะเริ่มต้น (init state) โปรแกรมจะไปอ่านข้อมูลในตารางย่อยที่เก็บข้อมูลของแพ็กเกตร้องขอเพื่อเชื่อมโยงเส้นทางตามการอ้างถึงของตัวชี้เพื่อนำข้อมูลของแพ็กเกตร้องขอเพื่อเชื่อมโยงเส้นทางมาเซตเป็นค่าตัวแปรทราฟฟิก แล้วเซตค่าอินเตอร์รัพท์โดยใช้รหัส (interrupt code) ให้ไปทำงานที่สถานะการสร้างแพ็กเกต (Generate state)

● สถานะการสร้างแพ็กเกต จะทำการสร้างแพ็กเกตและนำค่าตัวแปรต่าง ๆ ซึ่งอ่านออกมาในสถานะเริ่มต้นมาระบุให้กับแพ็กเกต โดยรูปแบบของแพ็กเกตและข้อมูลที่ส่งไปกับแพ็กเกตร้องขอการเชื่อมโยงเส้นทางแสดงดังรูปที่ 3.9

หมายเลขของโหนดต้นทางที่สร้างแพ็กเกตร้องขอการเชื่อมโยงเส้นทางจะถูกระบุลงในฟิลด์ SRC ID ทำการระบุหมายเลขโหนดปลายทางในฟิลด์ DEST ID ในฟิลด์ FORMAT จะระบุว่าเป็นแพ็กเกตควบคุม (signaling packet) ฟิลด์ TYPE จะระบุว่าเป็นแพ็กเกตร้องขอเพื่อเชื่อมโยงเส้นทาง (type call packet) ฟิลด์ ID จะระบุหมายเลขลำดับของแพ็กเกตร้องขอการเชื่อมโยงเส้นทาง ซึ่งจะใช้เป็นหมายเลขชี้เฉพาะของทราฟฟิก (traffic ID) หมายเลขนี้จะถูกระบุลงในแพ็กเกตตอบรับและแพ็กเกตข้อมูลด้วยเพื่อระบุว่าเป็นแพ็กเกตตอบรับและแพ็กเกตข้อมูลของ ทราฟฟิกใด และในฟิลด์ Information นั้นจะเตรียมไว้เพื่อใส่ความต้องการในการรับประกันคุณภาพอื่น ๆ ซึ่งข้อมูลในฟิลด์นี้สามารถเพิ่มเติมได้ตามความต้องการ ในการทดลองจะมีการระบุค่าช่องสัญญาณเวลาที่ต้องการจองในโหนดถัดไปลงในแพ็กเกตร้องขอการเชื่อมโยงเส้นทางด้วยและเมื่อมีการจองช่องสัญญาณเวลาซึ่งอาจจะเป็นตำแหน่งที่ระบุมากับแพ็กเกตหรือเป็นตำแหน่งอื่นสามารถจองได้ในช่วงของวินโดว์ และเมื่อจองช่องสัญญาณเวลาได้แล้วจะต้องทำการคำนวณเพื่อหาตำแหน่งที่น่าจะจองได้ในโหนดถัดไป แล้วเก็บค่าตำแหน่งนี้ไปกับแพ็กเกตร้องขอการเชื่อมโยงเส้นทางก่อนที่จะส่งแพ็กเกตนี้ออกไปยังโหนดถัดไป ซึ่งแพ็กเกตนี้จะถูกส่งไปเรื่อย ๆจนกระทั่งถึงโหนดปลายทางและจะรอรับแพ็กเกตตอบรับ (acknowledgement packet, ACK) ซึ่งถ้าเกิดปัญหาขึ้นระหว่างทาง เช่น กรณีที่มีทรัพยากรไม่เพียงพอระบบจะทำการปฏิเสธการร้องขอนั้นโดยส่งแพ็กเกตตอบรับเป็นแบบปฏิเสธ (non-acknowledgement packet, NACK) กลับมา แต่ถ้าระบบมีทรัพยากรเพียงพอและทำการสำรองช่องสัญญาณเวลาได้ตลอดทั้งเส้นทาง แพ็กเกตร้องขอการเชื่อมโยงเส้นทางนี้ก็จะถูกส่งถึงโหนดปลายทางได้ในที่สุดแล้วโหนดปลายทางจะสร้างแพ็กเกตตอบรับและส่งกลับมาตามเส้นทาง หากมีเส้นทางบางเส้นทางเสียหาย จะทราบได้โดยการตรวจสอบช่วงเวลารอคอยแพ็กเกตตอบรับ (time out) หากเกินเวลาแสดงว่ามีปัญหาเกิดขึ้นกับแพ็กเกตร้องขอการเชื่อมโยงเส้นทางหรือแพ็กเกตตอบรับ เราท์เตอร์ต้นทางจะต้องทำการส่งแพ็กเกตร้องขอการเชื่อมโยงเส้นทางไปใหม่ มีการเรียกใช้กระบวนการย่อยนี้อีกครั้งพร้อมกับค่าตัวชี้เดิมเพื่อนำข้อมูลของแพ็กเกตร้องขอการเชื่อมโยงเส้นทางเดิมมาใช้สร้างและส่งแพ็กเกตไปใหม่

<u>สถานะแยกแพ็กเกต (type state)</u>

เมื่อโหนดต้นทางหรือโหนดปลายทางได้รับแพ็กเกตเข้ามาจะพิจารณาที่ส่วนหัว (header) ของแพ็กเกตเป็นอย่างแรกเพื่อดูว่าเป็นแพ็กเกตประเภทไหน จากนั้นจึงจะแยกแพ็กเกตไปตามกระบวนการที่จะต้องจัดการกับแพ็กเกตประเภทนั้น และเก็บแพ็กเกตลงลิสต์ (List) ของกระบวน

การนั้น ๆ ซึ่งลิสต์ทำหน้าที่เหมือนเป็นบัฟเฟอร์เก็บแพ็กเกต หากไม่มีแพ็กเกตค้างในลิสต์เลย แพ็กเกตนี้ก็จะได้รับการพิจารณาทันที การพิจารณาจะเป็นไปตามกระบวนการของสถานะนั้น

สถานะได้รับแพ็กเกตร้องขอการเชื่อมโยงเส้นทาง (call_rcv state)

แพ็กเกตจะถูกเก็บลงลิสต์ของแพ็กเกตร้องขอการเชื่อมโยงเส้นทาง (call packet list) ซึ่งแพ็กเกตร้องขอการเชื่อมโยงเส้นทางนี้จะถูกจัดการเรียงตามลำดับการเข้ามาหรือตามลำดับในลิสต์นั่นเองโดยโปรแกรมจะนำแพ็กเกตออกจากลิสต์ แล้วทำการเข้าถึงข้อมูลในแพ็กเกต จากนั้นจึงจะนำข้อมูลมาพิจารณาเพื่อสร้างแพ็กเกตตอบรับซึ่งมีโครงสร้างของแพ็กเกตเช่นเดียวกับในรูป 3.9 แต่เซตค่าที่ TYPE เป็น ประเภทแพ็กเกตตอบรับ และระบุแอดเดรสของโหนดปัจจุบันลงในฟิลด์ SRC ID พร้อมทั้งระบุฟิลด์ DEST ID เป็นค่าแอดเดรสของโหนดต้นทางที่ส่งแพ็กเกตร้องขอการเชื่อมโยงเส้นทางมา จากนั้นจึงส่งแพ็กเกตตอบรับนี้ไปยังมอดูล sw เพื่อส่งแพ็กเกตกลับไปยังโหนดต้นทางต่อไป แพ็กเกตตอบรับนี้จะถูกส่งกลับไปตามเส้นทางเดิมเพื่อยืนยันการอนุมัติตลอดทั้งเส้นทาง

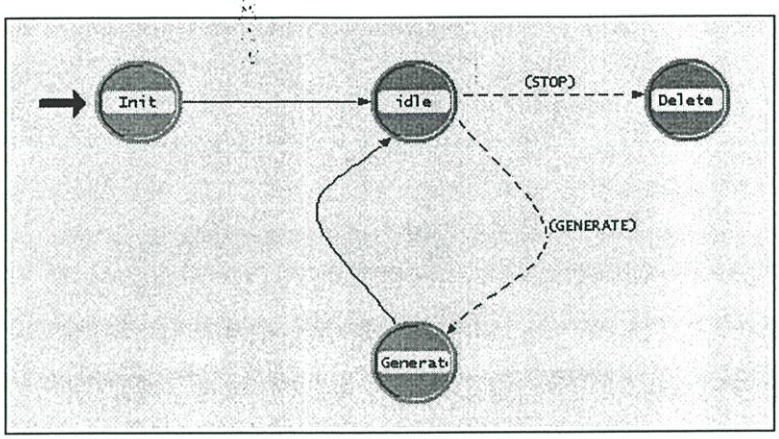
สถานะได้รับแพ็กเกตตอบรับ (ack_rcv state)

แพ็กเกตจะถูกนำออกมาจากลิสต์และเข้าถึงข้อมูลต่าง ๆ ซึ่งจะต้องพิจารณาก่อนว่าเป็นแบบตอบรับหรือปฏิเสธ ถ้าเป็นแบบปฏิเสธจะต้องดูหมายเลขลำดับของแพ็กเกตร้องขอการเชื่อมโยงเส้นทาง และทำการสร้างแพ็กเกตร้องขอการเชื่อมโยงเส้นทางใหม่อีกครั้ง โดยตรวจสอบหมายเลขลำดับของแพ็กเกตร้องขอการเชื่อมโยงเส้นทางที่ถูกส่งมากับแพ็กเกตตอบรับแบบปฏิเสธ และนำค่าหมายเลขลำดับนี้ไปหาลำดับของข้อมูลในตารางการสร้างแพ็กเกตร้องขอการเชื่อมโยงเส้นทาง และทำกระบวนการเรียกแพ็กเกตย่อยในการสร้างแพ็กเกตร้องขอการเชื่อมโยงเส้นทางอีกครั้งเมื่อสร้างแพ็กเกตร้องขอการเชื่อมโยงเส้นทางใหม่ได้แล้วก็จะส่งไปยังมอดูล sw เพื่อส่งออกและทำการร้องขอเพื่อเชื่อมโยงเส้นทางต่อไป หากแพ็กเกตที่ได้รับเป็นแบบตอบรับ จะพิจารณาว่าเป็นแพ็กเกตตอบรับของแพ็กเกตร้องขอการเชื่อมโยงเส้นทางหมายเลขลำดับใด จากนั้นจึงนำข้อมูลตัวแปรของแพ็กเกตร้องขอการเชื่อมโยงเส้นทางนั้นไปใช้ในการสร้างแพ็กเกตข้อมูลต่อไปโดยจะไปทำงานที่สถานะสร้างแพ็กเกตข้อมูล (gen_data state)

สถานะสร้างแพ็กเกตข้อมูล (gen_data state)

ที่สถานะนี้จะนำข้อมูลจากตารางข้อมูลของแพ็กเกตร้องขอการเชื่อมโยงเส้นทางไปเก็บในตารางคุณสมบัติในกระบวนการย่อยเพื่อสร้างแพ็กเกตข้อมูล ซึ่งจะมีการเซตค่าตัวชี้ตารางนี้พร้อม

กับการเรียกใช้กระบวนการย่อยในการสร้างแพ็กเกตข้อมูล (Data-generate child process) ซึ่งมีโครงสร้างการทำงานของสถานะต่าง ๆ ดังรูป 3.10 ซึ่งจะอธิบายในหัวข้อต่อไป

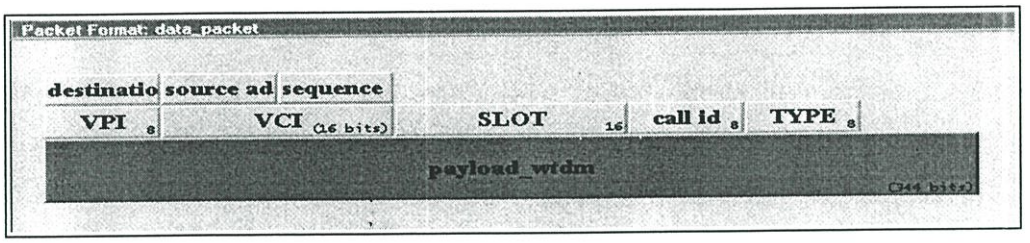


รูปที่ 3.10 แผนผังสถานะของกระบวนย่อยในการสร้างแพ็กเกตข้อมูล

● สถานะเริ่มต้น (init state) จะนำข้อมูลตัวแปรต่าง ๆ ของทราฟฟิกตามการอ้างของตัวชี้มาเก็บในตารางข้อมูลรายละเอียดทราฟฟิกในการสร้างแพ็กเกตข้อมูล เพื่อนำมาใช้ในกระบวนการสร้างแพ็กเกตข้อมูลต่อไป ซึ่งอัตราเร็วหรือระยะห่างในการสร้างแพ็กเกตข้อมูลจะขึ้นอยู่กับรูปแบบการกระจายของทราฟฟิกและคาบเวลาที่สุ่มเอาไว้ตั้งแต่แรก นำคาบเวลาและรูปแบบการกระจายมาหาเวลาในการสร้างแพ็กเกตข้อมูลและเซตเป็นเวลาในการอินเตอร์รัพท์เพื่อไปทำงานที่สถานะการสร้างแพ็กเกตข้อมูล (Generate data state) และเมื่อถึงเวลาหยุด โปรแกรมจะย้ายไปทำงานที่สถานะยกเลิกการสร้างแพ็กเกตและออกจากการทำงานของกระบวนการย่อย (delete state)

● สถานะการสร้างแพ็กเกตข้อมูล (Generate data state) จะทำการสร้างแพ็กเกตข้อมูล ซึ่งมีโครงสร้างของแพ็กเกตดังรูปที่ 3.11 แพ็กเกตจะถูกระบุค่าโหนดต้นทางลงในฟิลด์ source address ระบุค่าโหนดปลายทางลงในฟิลด์ destination address ระบุหมายเลขลำดับของแพ็กเกตข้อมูลลงในฟิลด์ data sequence ระบุประเภทของแพ็กเกตเป็นแพ็กเกตข้อมูลลงในฟิลด์ TYPE ระบุหมายเลขลำดับของทราฟฟิกลงในฟิลด์ call id ส่วนฟิลด์ VCI และ VPI และ SLOT นั้นจะถูกระบุค่าให้ที่อุปกรณ์เราท์เตอร์ เมื่อระบุข้อมูลลงในฟิลด์ต่าง ๆ แล้วจะส่งแพ็กเกตออกไปยังมอดูล sw และรอการอินเตอร์รัพท์ที่สถานะ idle เพื่อสร้างแพ็กเกตต่อไป

● สถานะยกเลิกกระบวนการย่อย (delete state) สถานะนี้จะทำการยกเลิกการสร้างแพ็กเกตข้อมูลและยกเลิกการทำงานของกระบวนการย่อยในการสร้างแพ็กเกตข้อมูล แล้วส่งแพ็กเกตเพื่อยกเลิกการใช้งานเส้นทาง (release packet) ออกไปยังมอดูล sw



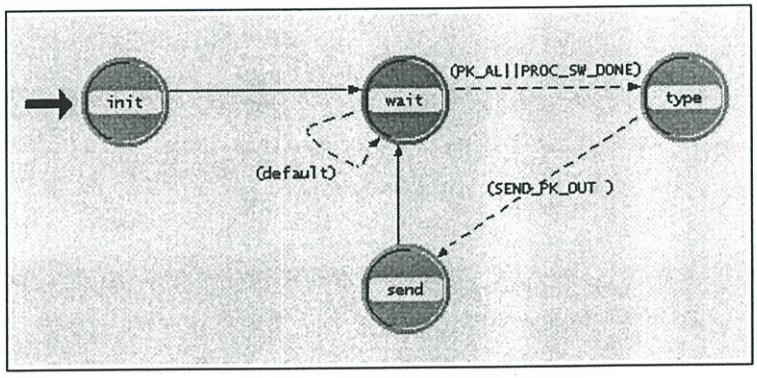
รูปที่ 3.11 โครงสร้างแพ็กเกตข้อมูล

สถานะรับแพ็กเกตข้อมูล (deata_rcv state)

กระบวนการในสถานะนี้จะนำแพ็กเกต ออกมาจากลิสต์และตรวจสอบหมายเลขลำดับของทราฟฟิก ทำการนับแพ็กเกตข้อมูลแยกตามหมายเลขลำดับของทราฟฟิกเพื่อหาอัตราการสูญหายของแพ็กเกตโดยจำนวนแพ็กเกตที่สูญหายกับแพ็กเกตที่ส่งออกของแต่ละทราฟฟิกแล้วนำมาหาค่าเฉลี่ยของการสูญหายแพ็กเกตข้อมูลรวมทั้งระบบ

3.2.1.3. **มอดูล** sw

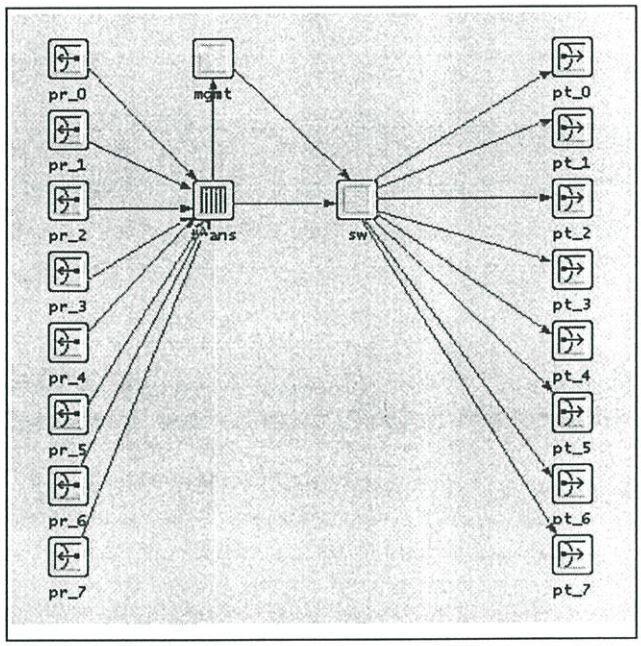
เมื่อได้รับแพ็กเกตเข้ามาจะเก็บแพ็กเกตลงลิสต์หากในลิสต์ไม่มีแพ็กเกตค้างอยู่ก็สามารถส่งแพ็กเกตออกได้ทันที



รูปที่ 3.12 แผนผังสถานะภายในมอดูล sw

การทำงานของมอดูล sw ที่อุปกรณ์สื่อสารปลายทางมีแผนผังการทำงานดังรูปที่ 3.12 โดยที่สถานะ init จะเป็นการเก็บข้อมูลและเซตค่าตัวแปลเริ่มต้น จากนั้นจะไปรอการอินเตอร์รัพท์ที่สถานะ wait จะมีเพียงอินเตอร์รัพท์แบบเหตุการณ์และเหตุการณ์ที่เข้ามาคือการมีแพ็กเกตเข้ามา แพ็กเกตทุกประเภทที่เข้านี้จะถูกส่งไปที่สถานะ type ซึ่งจะเก็บข้อมูลของแพ็กเกตในแต่ละทราฟฟิกก่อนที่จะส่งไปที่สถานะ send เพื่อส่งแพ็กเกตออกไปยังเราท์เตอร์ต้นทาง

3.2.2 อุปกรณ์เราท์เตอร์และสวิตช์จำลอง



รูปที่ 3.13 แบบจำลองภายในอุปกรณ์เราท์เตอร์และสวิตช์ (Router and switch node model)

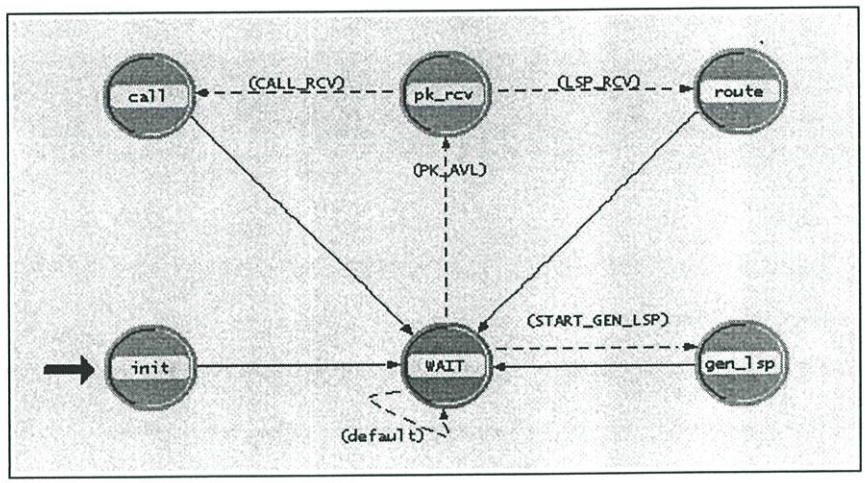
3.2.2.1 **มอดูล** trans

ในอุปกรณ์นี้ได้กำหนดให้มีพอร์ตขาเข้าและพอร์ตขาออกขนาด 8x8 ซึ่งพอร์ตขาเข้าจะเชื่อมต่อกับมอดูล trans ซึ่งแผนผังสถานะภายในมอดูล trans นี้จะเป็นดังรูปที่ 3.6 แต่การทำงานจะต่างจากมอดูล trans ที่อุปกรณ์สื่อสารปลายทาง โดยมอดูล trans ที่อุปกรณ์เราท์เตอร์หรือสวิตช์ทำหน้าที่แยกประเภทของแพ็กเกตที่เป็นแพ็กเกตควบคุมและแพ็กเกตข้อมูล ที่สถานะ svc_compl ซึ่งให้บริการส่งแพ็กเกตจะจัดการกับแพ็กเกตโดยส่งแพ็กเกตควบคุมไปยังมอดูล mgmt และเก็บลงลิสต์เพื่อดำเนินการต่อไป ส่วนแพ็กเกตข้อมูลจะถูกส่งไปเก็บลงลิสต์ที่มอดูล sw และรอดำเนินการต่อไปเช่นกัน

3.2.2.2. **มอดูล** mgmt

สถานะนี้เป็นสถานะที่จัดการกับแพ็กเกตควบคุมซึ่งมีทั้งที่เป็นแพ็กเกตร้องขอการเชื่อมโยงเส้นทาง แพ็กเกตตอบรับ และแพ็กเกตที่เกี่ยวข้องกับกระบวนการหาเส้นทาง ดังนั้นจึงมีการพัฒนาอัลกอริทึมของการหาเส้นทางและการจองช่องสัญญาณเวลาลงไปในมอดูลนี้ การทำงานที่มอดูล mgmt ของเราท์เตอร์และสวิตช์นั้นจะต่างกันโดยที่เราท์เตอร์มีกระบวนการค้นหา

เส้นทางแต่ที่สวิตช์ไม่มีกระบวนการค้นหาเส้นทาง ซึ่งจะอธิบายรายละเอียดต่อไป การทำงานในมอดูลนี้สามารถอธิบายเป็นแผนผังสถานะได้ดังรูปที่ 3.14 ซึ่งสถานะต่าง ๆ มีการทำงานดังนี้



รูปที่ 3.14 แผนผังสถานะภายในมอดูล mgmt

สถานะเริ่มต้น (init state)

ที่สถานะนี้จะทำการเซตค่าเริ่มต้นต่าง ๆ รวมทั้งสร้างและกำหนดหน่วยความจำให้กับลิสต์และตารางต่าง ๆ ที่จะต้องใช้ในมอดูลนี้ มีการติดตั้งและเก็บข้อมูลการเชื่อมโยงภายในระบบเครือข่ายจำลองซึ่งแบ่งออกเป็นขั้นตอนย่อยดังนี้

- Network configuration
- Neighbor configuration
- Node configuration

Network configuration

เพื่อให้ทราบข้อมูลของทั้งระบบ ทั้งค่าหมายเลขเฉพาะของทุกโหนด ค่าหมายเลขเฉพาะของลิงค์ที่ทำการเชื่อมโยงโหนดต่าง ๆ จึงออกแบบให้เก็บข้อมูลเริ่มต้นของระบบในสถานะนี้ ซึ่งระบบจริงนั้นบางโปรโตคอลไม่จำเป็นต้องรู้ข้อมูลและหมายเลขเฉพาะของทุกโหนด รู้เพียงหมายเลขเฉพาะของเราท์เตอร์หลักของเครือข่ายย่อย (Autonomous router) ก็สามารถส่งข้อมูลได้ แต่ในการทดลองนี้จำเป็นต้องรู้ข้อมูลและหมายเลขเฉพาะของทุกโหนดและทุกลิงค์เพื่อนำมาใช้อ้างอิงโหนดต้นทาง โหนดปลายทาง และลิงค์ที่ทำการเชื่อมโยง เนื่องจากการคำนวณเส้นทางต้องใช้ข้อมูลการใช้งานแบนด์วิดท์ในแต่ละโหนด และต้องปรับค่าภาระการใช้งานลิงค์เป็นระยะ

Neighbor configuration

ในขั้นตอนนี้จะทำให้ทราบหมายเลขเฉพาะของโหนดข้างเคียง หมายเลขเฉพาะของลิงค์ที่ทำการเชื่อมโยง พอร์ตรับและส่งระหว่างโหนดปัจจุบันกับโหนดข้างเคียงทุกโหนด ซึ่งข้อมูลนี้จะถูกเก็บลงในตารางและสามารถนำไปใช้ในกระบวนการต่าง ๆ ได้ กระบวนการที่ใช้ข้อมูลนี้ เช่น การกระจายแพ็กเกตแจ้งสถานะการใช้งานเส้นทางเชื่อมโยงไปยังโหนดอื่นที่ไม่ใช่โหนดที่ส่งมา

Node configuration

ในส่วนนี้จะมีการเซตค่าเกี่ยวกับสถานะการใช้งานช่องสัญญาณเวลาและกำหนดค่าเริ่มต้นต่าง ๆ กำหนดเวลาเริ่มต้นและคาบเวลาในการสร้างและส่งแพ็กเกตแจ้งสถานะการใช้งานเส้นทางเชื่อมโยง การสร้างและส่งแพ็กเกตร้องขออนุมัติเส้นทางเชื่อมโยง มีการสร้างและกำหนดหน่วยความจำให้ตารางและลิสต์ต่าง ๆ เช่น ตารางเส้นทาง ตารางวงจรเสมือน เป็นต้น

การกำหนดค่าสถานะการใช้งานช่องสัญญาณเวลานั้นจะทำการนับจำนวนพอร์ตขาออกที่ใช้งาน โดย 1 พอร์ตจะมี 1 เฟรมและใน 1 เฟรม จะถูกกำหนดให้มี 46 ช่องสัญญาณเวลา และแทนค่าสถานะการใช้งานช่องสัญญาณเวลาทั้งหมดด้วยบิตข้อมูล จากนั้นแปลงเป็นเลขฐาน 16 ซึ่งเริ่มต้นนั้นสถานะการใช้งานช่องสัญญาณเวลาจะเป็นสถานะว่างหรือถูกเป็น 1 ทั้งหมด ดังนั้นจะได้เลขฐาน 16 เป็น FFFFFFFFFFFC และเรียกชุดข้อมูลนี้ว่าชุดข้อมูลบิตแสดงสถานะการใช้งานเส้นทางเชื่อมโยง (Slot usage bit information, SUBI) และชุดข้อมูลนี้จะมีจำนวนเท่ากับจำนวนพอร์ตขาออกซึ่งเชื่อมต่ออยู่กับโหนดข้างเคียง และถูกเก็บค่าไว้ในตารางที่ 3.1 แล้วทำการเก็บข้อมูล SUBI ลงลิสต์เรียงตามลำดับหมายเลขพอร์ต เพื่อส่งข้อมูลลิสต์ไปพร้อมกับแพ็กเกตแจ้งสถานะการใช้งานเส้นทางเชื่อมโยง ดังนั้นโหนดต่าง ๆ จะรู้ข้อมูลสถานะการใช้งานช่องสัญญาณเวลาในเฟรมของทุกพอร์ต

ตารางที่ 3.1 ตารางเก็บข้อมูลการเชื่อมโยงกับโหนดข้างเคียง

Neighbor_Info_Table		คำอธิบาย
Objid	neighbor_id	หมายเลขเฉพาะของโหนดข้างเคียง
int	link_id	หมายเลขเฉพาะของลิงค์ที่เชื่อมกับโหนดข้างเคียง
int	port_id	หมายเลขเฉพาะของพอร์ต
char*	subi	ข้อมูลสถานะการใช้งานช่องสัญญาณเวลา

การกำหนดเวลาเริ่มต้นสร้างและส่งแพ็กเกตแจ้งสถานะการใช้งานเส้นทางเชื่อมโยงนั้นจะใช้การสุ่มค่าทางเวลาเพื่อเลือกเวลาเริ่มต้นให้แต่ละโหนดส่วนคาบเวลาในการสร้างและส่งแพ็กเกตนั้นสามารถกำหนดได้ที่ตารางคุณสมบัติ ซึ่งผู้ใช้สามารถกำหนดและเปลี่ยนแปลงค่านี้ได้ ซึ่งใน

วิธีการหาเส้นทางแบบ OSPF นั้นโดยทั่วไปจะเซตค่า default ไว้ที่ 5000 มิลลิวินาที ค่านี้จะถูกนำไปกำหนดเวลาในการสร้างแพ็กเกตที่สถานะ gen_lsp โดยใช้ self interrupt

สถานะเริ่มต้นนี้เมื่อทราบข้อมูลลิงค์ทุกลิงค์ในระบบ จะคำนวณหาค่าภาระการใช้งานลิงค์ ซึ่งในงานวิจัยนี้ได้กำหนดให้เป็นดังสมการที่ 3.10 เวลาหน่วงเนื่องจากการส่งผ่านเส้นทางเชื่อมโยงนั้นคำนวณโดยอัตราส่วนของความยาวของลิงค์ต่อความเร็วในการส่งผ่านลิงค์ ซึ่งความยาวของลิงค์สามารถระบุได้ที่ตารางคุณสมบัติของลิงค์ โดยทั่วไประยะระหว่างแต่ละ Hop จะไม่เกิน 2 กิโลเมตร ตามมาตรฐานของการส่งผ่านสายใยแก้วแบบมัลติโหมด และความเร็วที่ส่งผ่านสายใยแก้วจะเป็นความเร็วแสงซึ่งมีค่าเท่ากับ $3x10^8$ เมตรต่อวินาที ส่วนจำนวนช่องสัญญาณเวลาที่ซิงโครนัสเริ่มต้นจะเท่ากับ 46 ช่องสัญญาณเวลาเพราะยังไม่มีการใช้งานช่องสัญญาณเวลาใด เมื่อมีการปรับค่าภาระการใช้งานลิงค์แล้วจะคำนวณหาเส้นทางโดยใช้หลักการ SPF แล้วทำการสุ่มเพื่อเลือกช่องสัญญาณเวลาที่สามารถใช้งานได้ต่อไป เมื่อได้เส้นทางและช่องสัญญาณเวลาแล้วจะเก็บลงในตารางเส้นทางดังตารางที่ 3.2 และเมื่อมีการรับแพ็กเกตแจ้งสถานะการใช้งานเส้นทางเชื่อมโยงเข้ามาจึงจะทำการค้นหาเส้นทางใหม่และเก็บข้อมูลลงในตารางอีกครั้ง

ตารางที่ 3.2 ตารางเส้นทาง

Route_Table		คำอธิบาย
Objid	Destination_id	หมายเลขเฉพาะของโหนดปลายทาง
Route*	Route_ptr	ตัวชี้เส้นทางอ้างถึงเส้นทางที่จะไปยังโหนดปลายทาง
int	Slot_alloc	ช่องสัญญาณเวลาที่เหมาะสม
int*	bit_map_result	ชุดข้อมูลบิตที่ทำการเทียบบิตทั้งเส้นทางแล้ว

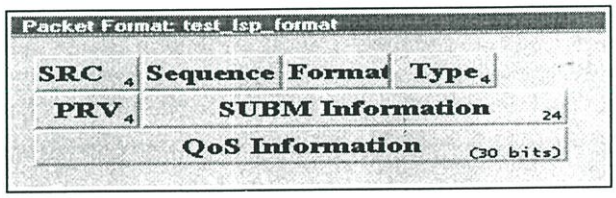
การสร้างตารางจัดลำดับแพ็กเกตเพื่อส่งออก และสร้างบัฟเฟอร์เพื่อเก็บแพ็กเกตในแต่ละช่องสัญญาณเวลาในเฟรม พอร์ตขาเข้าแต่ละพอร์ตจะมีตารางจัดลำดับแพ็กเกตเพื่อส่งออก และตารางจัดลำดับแพ็กเกตเพื่อส่งออกนั้นจะมีขนาดเท่ากับขนาดของเฟรม คือ 46 ตำแหน่ง ในแต่ละตำแหน่งจะเชื่อมโยงกับตารางข้อมูลซึ่งประกอบด้วยหมายเลขโหนดต้นทาง หมายเลขลำดับเฉพาะของทราฟฟิก ค่า VCI ที่ใช้แลกเปลี่ยนและมีลิสต์ตัวชี้ไปยังบัฟเฟอร์ขาออกซึ่งเป็นลิสต์ในการเก็บแพ็กเกตก่อนที่จะส่งออกไป ซึ่งบัฟเฟอร์นี้จะมีขนาดไม่จำกัดและหนึ่งช่องสัญญาณเวลาจะมี 1 ลิสต์ เนื่องจากกำหนดให้หนึ่งช่องสัญญาณเวลาสามารถบริการได้หนึ่งทราฟฟิก เมื่อทำการเซตค่าเริ่มต้นต่าง ๆ แล้วก็จะไปรออยู่ที่สถานะ wait เพื่อรอเวลาอินเตอร์รัพท์หรือเหตุการณ์อินเตอร์รัพท์ต่อไป

ตารางที่ 3.3 ตารางจัดลำดับแพ็กเกตเพื่อส่งออก

TR_Des_Table		คำอธิบาย
Objid	Source_id	หมายเลขเฉพาะของโหนดต้นทางที่ส่งแพ็กเกต
int	Call_id	หมายเลขลำดับเฉพาะของทราฟฟิก
int	VCI_IN	ค่า VCI ที่สัมพันธ์กับโหนดก่อนหน้า
int	VCI_OUT	ค่า VCI ที่สัมพันธ์กับโหนดถัดไป
List*	Buffer_Iptr	ตัวชี้บัฟเฟอร์ที่อ้างถึงบัฟเฟอร์ที่เก็บแพ็กเกตเพื่อรอส่งออก

● สถานะรอการอินเตอร์รัพท์ (wait state) การอินเตอร์รัพท์นี้อาจจะเป็นอินเตอร์รัพท์โดยการตั้งเวลาหรืออินเตอร์รัพท์โดยเหตุการณ์ก็ได้ ในสถานะนี้การอินเตอร์รัพท์มี 2 แบบคือ การตั้งเวลาเพื่อสร้างแพ็กเกตแจ้งสถานะการใช้งานเส้นทางเชื่อมโยง และรอเหตุการณ์ที่มีแพ็กเกตควบคุมเข้ามา

● สถานะการสร้างแพ็กเกตแจ้งสถานะการใช้งานเส้นทางเชื่อมโยง (gen_lsp state) ที่สถานะนี้จะทำการสร้างแพ็กเกตแจ้งสถานะการใช้งานเส้นทางเชื่อมโยง ซึ่งมีโครงสร้างของแพ็กเกตดังรูปที่ 3.15 ซึ่งข้อมูลสถานะการใช้งานช่องสัญญาณเวลาจะถูกเก็บเป็นลิสต์ของ ชุดข้อมูล SUBI และลิสต์นี้จะถูกส่งไปพร้อมกับแพ็กเกตแจ้งสถานะการใช้งานเส้นทางเชื่อมโยงโดยเก็บข้อมูลลงในฟิลด์ SUBM Information



รูปที่ 3.15 โครงสร้างแพ็กเกตแจ้งสถานะการใช้งานเส้นทางเชื่อมโยง

● กรณีที่มีแพ็กเกตเข้ามา แพ็กเกตจะถูกแยกประเภทที่สถานะ pk_rcv และเก็บลงลิสต์ตามประเภทของแพ็กเกต แพ็กเกตควบคุมที่เข้ามาที่มอดูล mgmt นี้จะมีแพ็กเกตร้องขอการเชื่อมโยงเส้นทาง แพ็กเกตตอบรับและแพ็กเกตแจ้งสถานะการใช้งานเส้นทางเชื่อมโยง

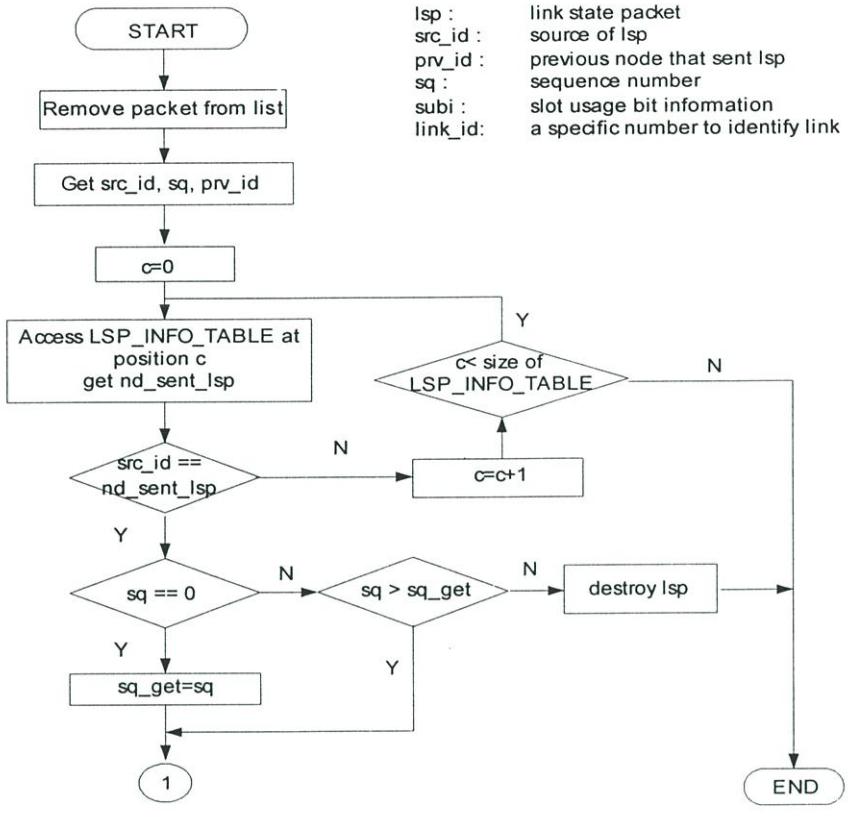
● ซึ่งเมื่อมีแพ็กเกตแจ้งสถานะการใช้งานเส้นทางเชื่อมโยงเข้ามาจะถูกส่งไปยังสถานะ route ที่สถานะนี้ จะเก็บข้อมูลต่าง ๆ ลงในตารางข้อมูลแพ็กเกตแจ้งสถานะการใช้งานเส้นทางเชื่อมโยง โดยแยกตามแอดเดรสของโหนดต้นทางที่ส่งแพ็กเกตมาดังตารางที่ 3.4

ตารางที่ 3.4 ตารางข้อมูลแพ็กเกตแจ้งสถานะการใช้งานเส้นทางเชื่อมโยง

LSP_INFO_TABLE	คำอธิบาย
Objid nd_sent_lsp;	โหนดต้นทางของแพ็กเกต
Int sq;	หมายเลขลำดับ
List* sq_info_nb_nd_lptr;	ตัวชี้ลิสต์เก็บโหนดข้างเคียงของ nd_sent_lsp
List* sq_info_end_nd_lptr;	ตัวชี้ลิสต์เก็บโหนดข้างเคียงของ nd_sent_lsp ที่เป็นโหนดสื่อสารปลายทาง
List* sq_info_subm_get_lptr;	ตัวชี้ลิสต์เก็บข้อมูลสถานะการใช้งานช่องสัญญาณเวลา

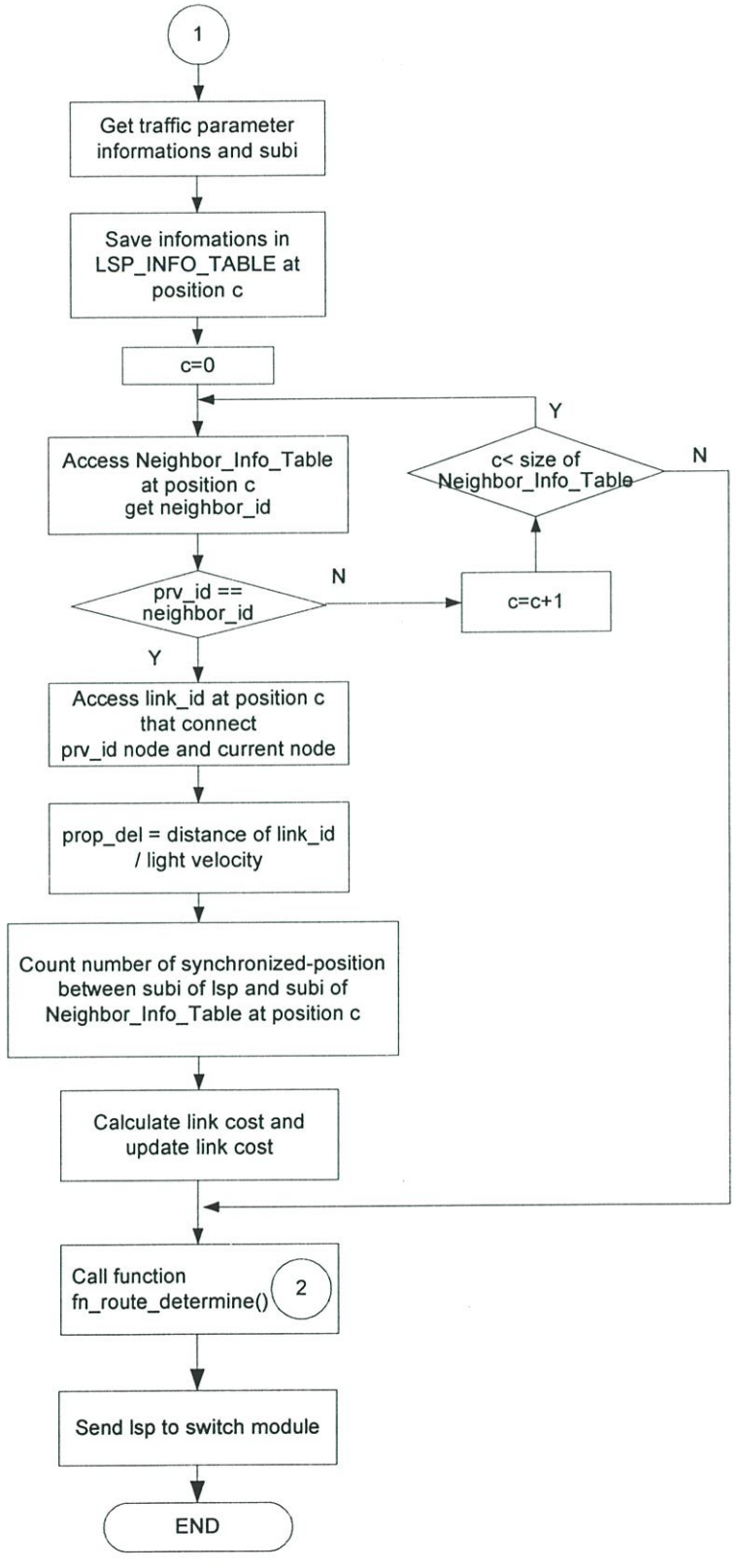
การส่งแพ็กเกตแจ้งสถานะการใช้งานเส้นทางเชื่อมโยงนั้นจะเป็นแบบกระจายซึ่งกระบวนการกระจายแพ็กเกตออกไปทางพอร์ตต่าง ๆ นั้นมีความเป็นไปได้ที่แพ็กเกตซึ่งมาจากโหนดเดียวกันนั้นอาจจะได้รับเข้ามาซ้ำกันได้ ดังนั้นจึงต้องมีการกำหนดหมายเลขลำดับเฉพาะให้สำหรับแต่ละแพ็กเกต หากเป็นหมายเลขซ้ำจะต้องทำการลบแพ็กเกตทิ้ง เพื่อไม่ให้เป็นการส่งกระจายต่อออกไปและไม่ต้องทำการปรับค่าข้อมูลด้วย ส่วนลิสต์ที่มีในตารางข้อมูลแพ็กเกตแจ้งสถานะการใช้งานเส้นทางเชื่อมโยงนั้น จะเก็บข้อมูลการเชื่อมโยงกับโหนดข้างเคียงของโหนดต้นทางที่ส่งแพ็กเกตมาซึ่งประกอบด้วย หมายเลขของโหนดข้างเคียงซึ่งจะเก็บลงลิสต์เรียงตามลำดับหมายเลขพอร์ต หมายเลขลำดับของโหนดข้างเคียงที่เป็นโหนดสื่อสารปลายทาง และสถานะการใช้งานช่องสัญญาณเวลา ซึ่งข้อมูลโหนดที่ทำการเชื่อมโยงอยู่นี้อาจมีการเปลี่ยนแปลงเนื่องจากการขยายของระบบเครือข่ายได้และข้อมูลที่เปลี่ยนแปลงนี้จะมีการแจ้งเพื่อปรับข้อมูลปัจจุบันตามระยะการส่งแพ็กเกตแจ้งสถานะการใช้งานเส้นทางเชื่อมโยง

จากแผนผังในรูปที่ 3.16 เมื่อได้รับแพ็กเกตแจ้งสถานะการใช้งานเส้นทางเชื่อมโยงจะทำการเข้าถึงข้อมูลในแพ็กเกตเพื่อพิจารณาว่าเป็นแพ็กเกตที่ถูกส่งมาจากโหนดใด จากนั้นจะไปดูลำดับหมายเลขของแพ็กเกต ถ้าเป็นหมายเลขที่น้อยกว่าหมายเลขที่เคยได้รับและเก็บข้อมูลไว้แล้วแพ็กเกตนั้นจะถูกลบทิ้ง แต่ถ้าเป็นลำดับหมายเลขที่มากกว่าจะต้องทำการเก็บข้อมูลใหม่ลงในตาราง LSP_INFO_TABLE

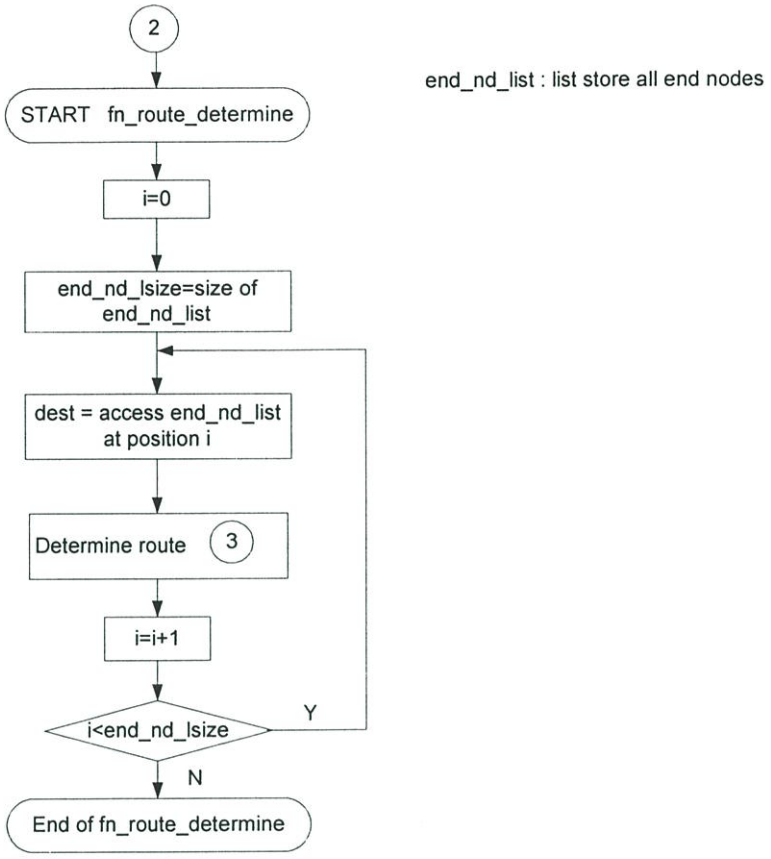


รูปที่ 3.16 โฟล์วชาร์ตปรับปรุงข้อมูลเมื่อรับแพ็กเกตแจ้งสถานะการใช้งานเส้นทางเชื่อมโยง

แผนผังในรูปที่ 3.17 เมื่อเก็บข้อมูลลงในตารางแล้วก็จะทำการพิจารณาว่าโหนดที่ส่งแพ็กเกตมานี้เป็นโหนดข้างเคียงหรือไม่ ถ้าเป็นโหนดข้างเคียงจะทำการปรับค่าภาระการใช้งานลิงค์แล้วจึงทำการคำนวณหาเส้นทางแต่ถ้าไม่ใช่โหนดข้างเคียงก็ไม่ต้องปรับค่าภาระการใช้งานลิงค์เนื่องจากปัจจัยที่กระทบต่อค่าภาระการใช้งานลิงค์คือจำนวนช่องสัญญาณเวลาที่ซิงโครนัสซึ่งจะพิจารณาระหว่างสองโหนดใด ๆ ที่อยู่ติดกัน ดังนั้นหากมีการเปลี่ยนแปลงสถานะการใช้งานช่องสัญญาณเวลาของโหนดข้างเคียงจะมีผลกระทบต่อค่าภาระการใช้งานลิงค์ นั่นคือทำให้ ค่าภาระการใช้งานลิงค์ที่เชื่อมต่ออยู่กับโหนดข้างเคียงเปลี่ยนไป เมื่อปรับปรุงข้อมูลเสร็จแล้วจึงเรียกใช้ฟังก์ชันการค้นหาเส้นทางเพื่อทำการคำนวณหาเส้นทาง ซึ่งในฟังก์ชันนี้จะมีสองส่วน โดยส่วนแรกจะเป็นการหาเส้นทางที่สั้นที่สุดซึ่งพิจารณาจากค่าภาระการใช้งานลิงค์และเมื่อได้เส้นทางที่สั้นที่สุดจำนวนหนึ่งแล้วจะทำการนำเส้นทางที่มีค่าภาระการใช้งานลิงค์น้อยที่สุดและมีจำนวน Hob น้อยที่สุดมาพิจารณาเทียบบิตก่อน การหาเส้นทางที่สั้นที่สุดนั้นจะแสดงในรูปที่ 3.19 และเมื่อหาเส้นทางที่สั้นที่สุดและมีจำนวน Hop น้อยที่สุดได้แล้วจะทำการเทียบบิตเพื่อเลือกเส้นทางที่เหมาะสมและคำนวณตำแหน่งที่น่าจะจองต่อไป



รูปที่ 3.17 โฟล์วชาร์ตปรับปรุงข้อมูลเมื่อรับแพ็กเกตแจ้งสถานะการใช้งานเส้นทางเชื่อมโยง (ต่อ)

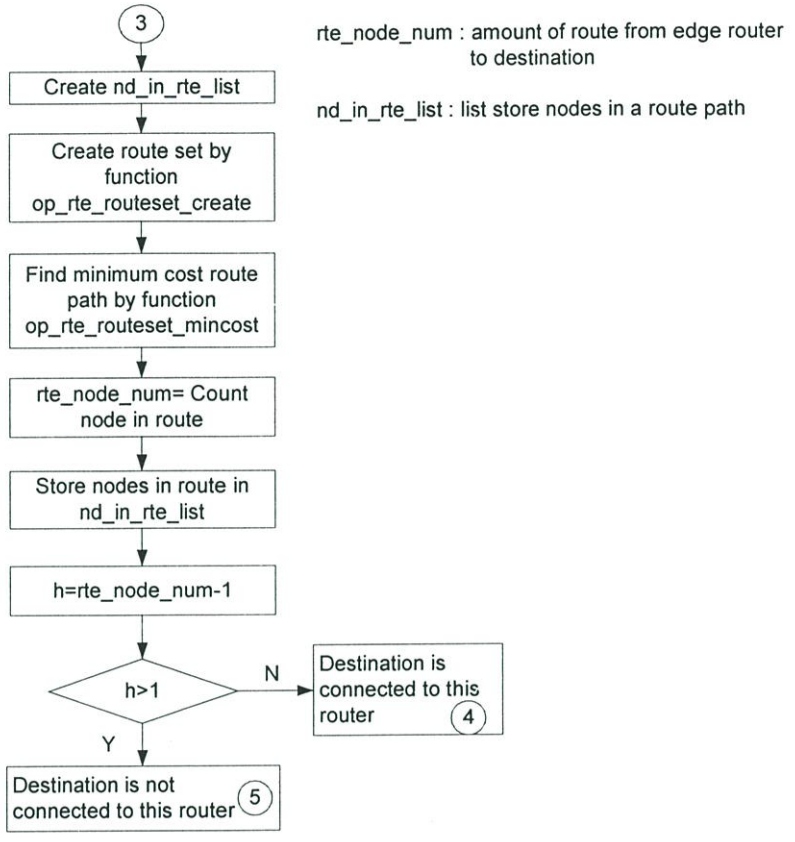


รูปที่ 3.18 โฟล์วชาร์ตฟังก์ชันการค้นหาเส้นทาง

ฟังก์ชันการค้นหาเส้นทางนี้จะถูกเรียกใช้ทุกครั้งที่มีการปรับค่าข้อมูลเนื่องจากได้รับแพ็กเกตแจ้งสถานะการใช้งานเส้นทางเชื่อมโยง รวมทั้งกรณีที่มีการเปลี่ยนแปลงขนาดของระบบหรือมีการเชื่อมต่ออุปกรณ์เพิ่มในระบบเพราะต้องคำนวณหาเส้นทางสำหรับโหนดปลายทางใหม่ การพิจารณาเส้นทางจะทำการหาเส้นทางสำหรับโหนดสื่อสารปลายทางทุกโหนด ซึ่งโหนดสื่อสารปลายทางจะถูกเก็บค่าหมายเลขเฉพาะไว้ในลิสต์ end_nd_list ตั้งแต่สถานะเริ่มต้นการทำงานและจะมีการปรับค่าในลิสต์เมื่อมีการเปลี่ยนแปลงขนาดของระบบทุกครั้ง ดังนั้นการค้นหาเส้นทางจึงต้องเข้าถึงทุกตำแหน่งในลิสต์นี้ และนำค่าหมายเลขเฉพาะที่ได้มาเซตเป็นโหนดปลายทาง แล้วทำการหาเส้นทางไปยังปลายทางดังกล่าว ซึ่งในรูปที่ 3.18 จะเห็นว่าฟังก์ชันนี้จะทำการหาเส้นทางให้กับทุกเส้นทางที่มีในลิสต์ ส่วนที่เป็นการคำนวณเพื่อค้นหาเส้นทางและการเทียบบิตนั้นจะอธิบายในโฟล์วชาร์ตย่อยต่อไป

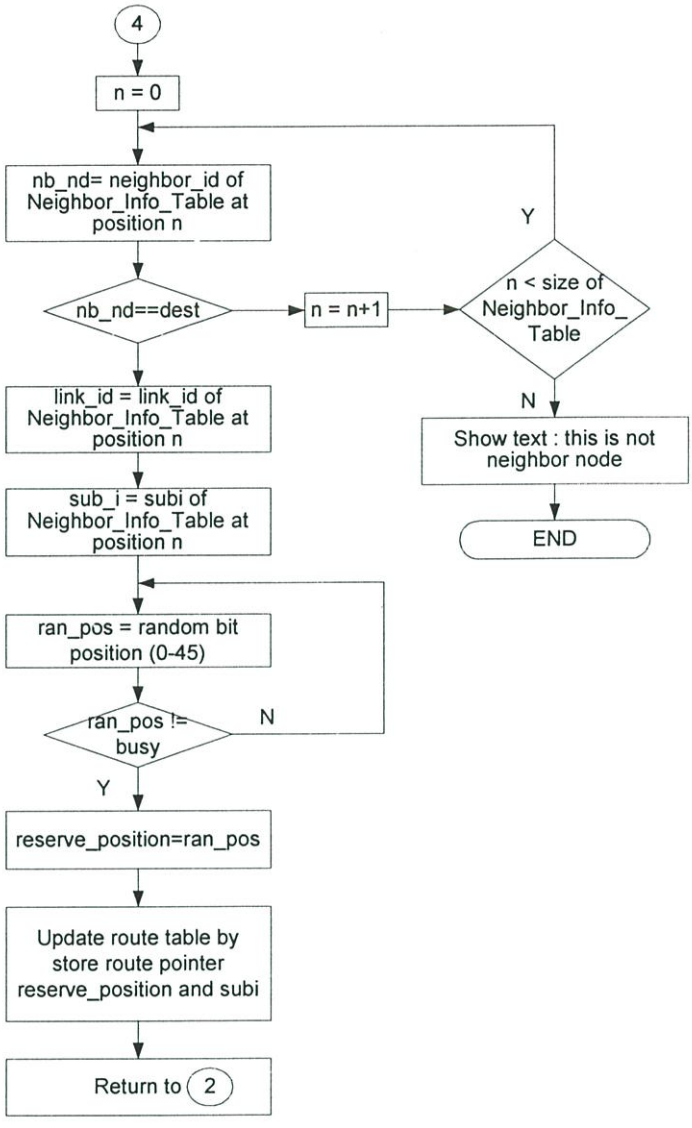
โฟล์วชาร์ตย่อยในรูปที่ 3.19 จะใช้ฟังก์ชันการหาเส้นทางที่สั้นที่สุดโดยใช้ฟังก์ชันที่มีในโปรแกรมจำลองเครือข่าย Opnet สร้างเส้นทางที่เป็นไปได้ไปยังปลายทางใด ๆ โดยใช้ฟังก์ชัน op_rte_routeset_create () ซึ่งผลที่ได้จากฟังก์ชันนี้จะเป็นเซตของเส้นทางกลุ่มหนึ่งจากเราท์เตอร์ต้นทางไปยังปลายทางที่เลือก จากนั้นจึงใช้ฟังก์ชัน op_rte_routeset_mincost () เพื่อทำการเลือก

เส้นทางที่มีค่าภาระการใช้ลิงค์น้อยที่สุด ซึ่งค่าภาระการใช้ลิงค์นี้ถูกคำนวณและเซตค่าให้ทุกครั้งที่มีการปรับค่าสถานะการใช้งานเส้นทางเชื่อมโยงใหม่



รูปที่ 3.19 โฟล์วชาร์ตฟังก์ชันการค้นหาเส้นทาง (ต่อ)

เมื่อได้เส้นทางแล้วจะทำการหาจำนวนโหนดที่มีในเส้นทาง (rte_node_num) ซึ่งจะเริ่มนับตั้งแต่เราท์เตอร์ต้นทางที่ทำการค้นหาเส้นทาง ไปจนถึงโหนดปลายทางดังกล่าว ดังนั้นเมื่อคิดจากเราท์เตอร์ต้นทางถึงเราท์เตอร์ปลายทางที่เชื่อมโยงโดยตรงกับโหนดปลายทางนั้นจะมีจำนวนโหนดเท่ากับ h และเมื่อพิจารณาค่า h ที่น้อยกว่าหรือเท่ากับ 1 แสดงว่ามีเพียงสองโหนด นั่นคือเราท์เตอร์ต้นทางและโหนดปลายทางดังนั้นโหนดปลายทางจะเป็นโหนดที่เชื่อมต่อกับเราท์เตอร์โดยตรงหรือเป็นโหนดข้างเคียงของเราท์เตอร์นั่นเอง แต่ถ้า h มากกว่า 1 แสดงว่าเป็นโหนดปลายทางที่ต้องส่งผ่านระบบและมีจำนวน hop ในเส้นทางมากกว่า 1 hop ซึ่งขั้นตอนการพิจารณากรณีทั้งสองนี้จะต่างกันไป



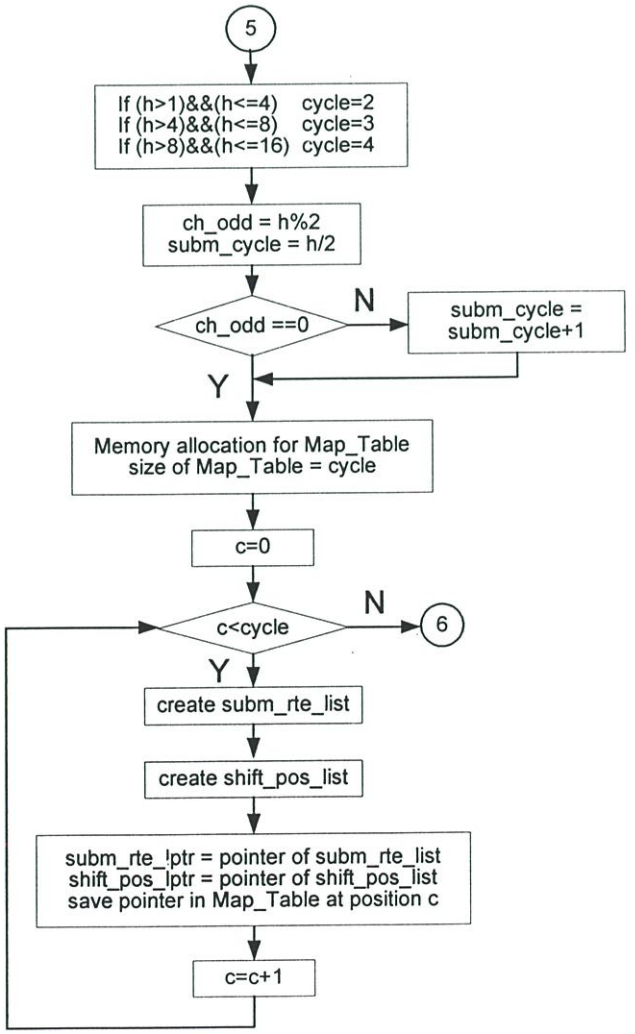
รูปที่ 3.20 โฟล์วชาร์ตฟังก์ชันการค้นหาเส้นทาง (ต่อ)

โฟล์วชาร์ตดังรูปที่ 3.20 เป็นกรณีที่โหนดปลายทางเป็นโหนดที่อยู่ติดกับเราท์เตอร์ต้นทางโดยตรงหรือเป็นโหนดข้างเคียงของเราท์เตอร์ต้นทาง การหาตำแหน่งช่องสัญญาณเวลาที่เหมาะสมในพอร์ตที่จะออกไปยังโหนดปลายทางดังกล่าวจึงสามารถพิจารณาได้จากข้อมูลการใช้งานช่องสัญญาณเวลาภายในโหนดเอง ซึ่งข้อมูลดังกล่าวนี้ได้ถูกเก็บไว้ในตาราง Neighbor_Info_Table และเมื่อได้ชุดข้อมูลสถานะการใช้งานช่องสัญญาณเวลาแล้วจะทำการสุ่มเพื่อเลือกช่องสัญญาณเวลาที่เป็นสถานะว่าง การสุ่มนี้จะดีกว่าการเลือกตำแหน่งแบบเรียงลำดับ เนื่องจากความต้องการใช้งานตำแหน่งที่ซิงโครนัสของทราฟฟิกอื่น ๆ จะได้รับความยืดหยุ่นในการสำรองช่องสัญญาณเวลามากกว่าการจองแบบเรียงลำดับ จากนั้นจะทำการเก็บเส้นทางและช่องสัญญาณเวลาที่สามารถจองได้ไว้ในตารางเส้นทาง เมื่อเก็บเส้นทางแล้วจะทำการหาเส้นทาง

สำหรับโหนดปลายทางถัดไปโดยกลับไปทำงานดังโฟล์วชาร์ตรูปที่ 3.18 กรณีที่แพ็กเกตร้องขอเส้นทางเชื่อมโยงเข้ามาและใช้งานช่องสัญญาณวลานี้ไปแล้วและยังไม่มีการคำนวณหาเส้นทางใหม่ก็ให้ใช้เส้นทางเดิมแต่ทำการเลือกสุ่มค่าตำแหน่งใหม่แล้วเก็บค่าลงในตารางเส้นทางอีกครั้งซึ่งจะอธิบายในสถานะ call ต่อไป

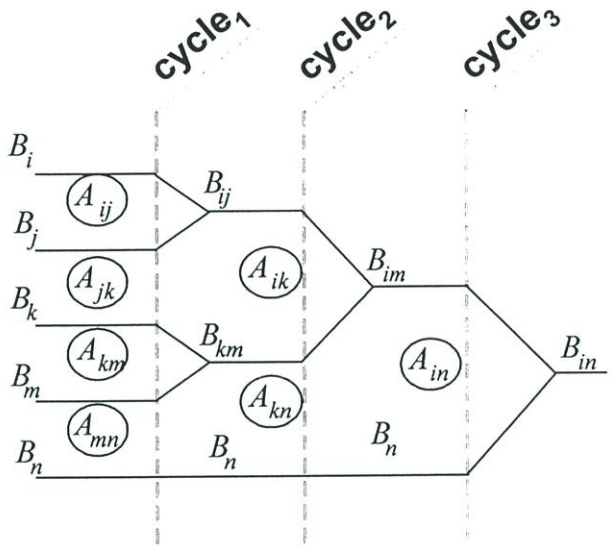
ตารางที่ 3.5 ตารางข้อมูลแพ็กเกตแจ้งสถานะการใช้งานเส้นทางเชื่อมโยง

Map_Table	คำอธิบาย
List* subm_rte_lptr	ตัวชี้ลิสต์ซึ่งอ้างถึงลิสต์ที่เก็บค่า SUBI ในการเทียบบิตแต่ละรอบ
List* shift_pos_lptr	ตัวชี้ลิสต์ซึ่งอ้างถึงลิสต์ที่เก็บตำแหน่งเยื้องในการเทียบบิตแต่ละรอบ



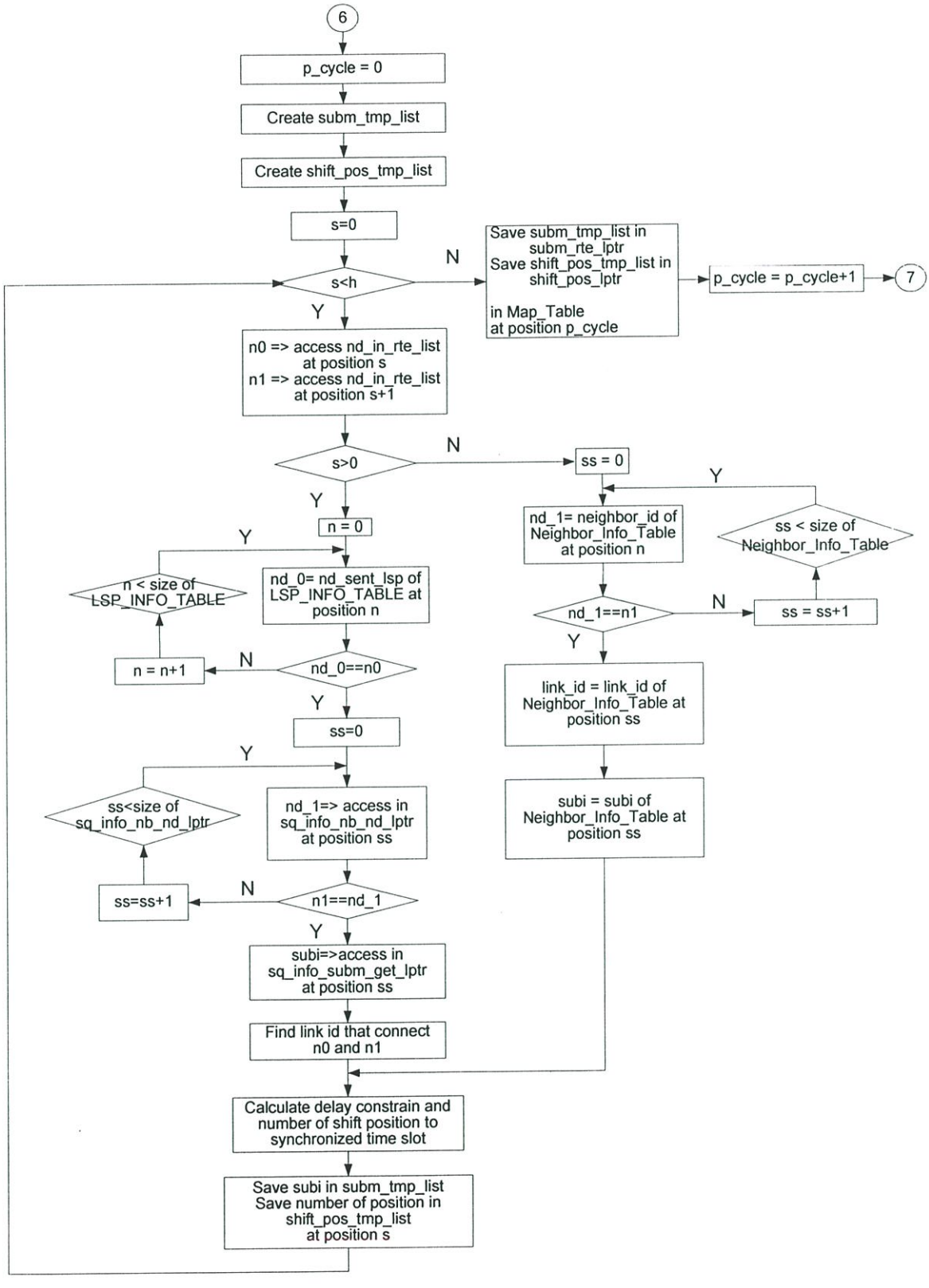
รูปที่ 3.21 โฟล์วชาร์ตฟังก์ชันการค้นหาเส้นทาง (ต่อ)

ในการเทียบบิตโดยการ AND จะใช้วิธีการเทียบบิตดังรูปที่ 3.4 ซึ่งทำการ AND ชุดข้อมูลทีละคู่ โดยจะสร้างตารางและลิสต์เพื่อเก็บชุดบิตข้อมูลและตำแหน่งเยื้องระหว่างโหนดแต่ละคู่ ในรูปที่ 3.21 นั้นจะเป็นเพียงการกำหนดหน่วยความจำให้กับตารางและเป็นการสร้างลิสต์ไว้เพื่อเก็บข้อมูลในการคำนวณเท่านั้น ส่วนการเทียบบิตนั้นสามารถอธิบายได้ดังรูปที่ 3.22

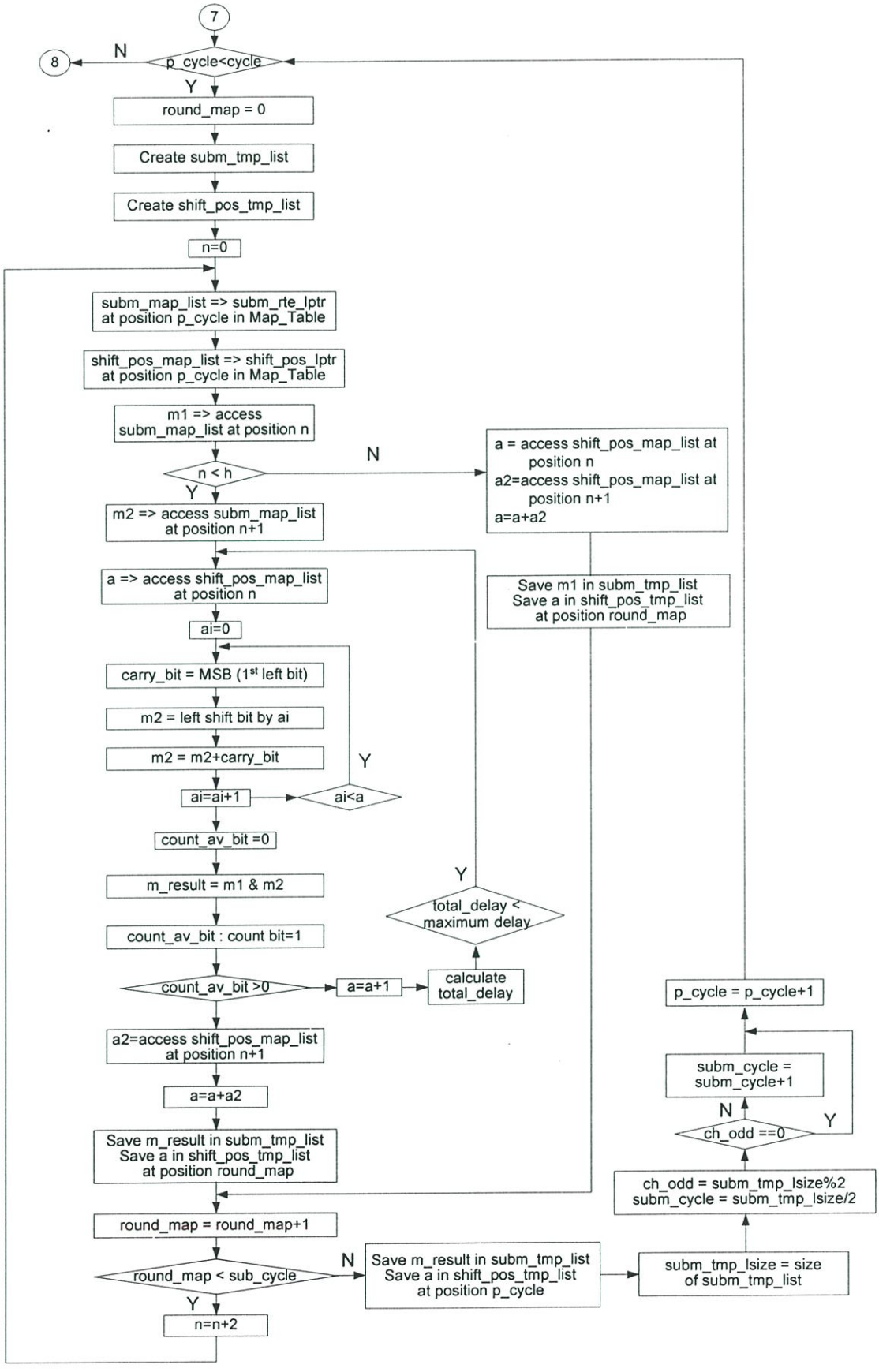


รูปที่ 3.22 ตัวอย่างการเก็บข้อมูลสถานะการใช้งานช่องสัญญาณเวลาและตำแหน่งเยื้องในการเทียบบิตแต่ละรอบ

วิธีการเทียบบิตจะต้องทำการพิจารณาจำนวนรอบที่จะทำการ AND จากรูปที่ 3.22 จำนวนรอบหรือ cycle มี 3 รอบดังนั้นขนาดของ Map_Table จะถูกกำหนดหน่วยความจำให้จะมีขนาดเท่ากับ 3 ในตำแหน่งแรกของตาราง Map_table นั้นจะเก็บตัวชี้ลิสต์ subm_rte_list ซึ่งจะเก็บข้อมูลบิต B_i B_j B_k B_m และ B_n และจะเก็บตัวชี้ลิสต์ shift_pos_list ที่เก็บตำแหน่งเยื้อง A_{ij} A_{jk} A_{km} และ A_{mn} การเก็บข้อมูลในการเทียบบิตรอบแรกนี้จะเป็นดังโฟล์วชาร์ตในรูปที่ 3.23 แล้วจึงทำการ AND ข้อมูลเป็นคู่ จากรูปที่ 3.22 จะแบ่งการ AND ออกเป็น 3 ส่วน ส่วนแรก B_i AND B_j โดยใช้ตำแหน่งเยื้อง A_{ij} ผลที่ได้คือ B_{ij} ส่วนที่สอง B_k AND B_m โดยใช้ตำแหน่งเยื้อง A_{km} ผลที่ได้คือ B_{km} และส่วนสุดท้ายจะเป็น B_n ซึ่งไม่มีคู่ในการ AND ดังนั้นการเก็บข้อมูลในรอบที่สองจะประกอบด้วย B_{ij} B_{km} และ B_n ค่าตำแหน่งเยื้องที่เก็บจะประกอบด้วย $A_{ik} = A_{ij} + A_{jk}$ และ $A_{kn} = A_{km} + A_{mn}$ กรณีที่ผลการ AND ไม่มีช่องสัญญาณเวลาที่ใช้ได้ จะต้องทำการเลื่อนบิตเพิ่มและทุกครั้งที่เลื่อนบิตต้องคำนวณเวลาหน่วงที่เปลี่ยนไปซึ่งกระบวนการนี้ได้แสดงไว้ดังโฟล์วชาร์ตในรูปที่ 3.24



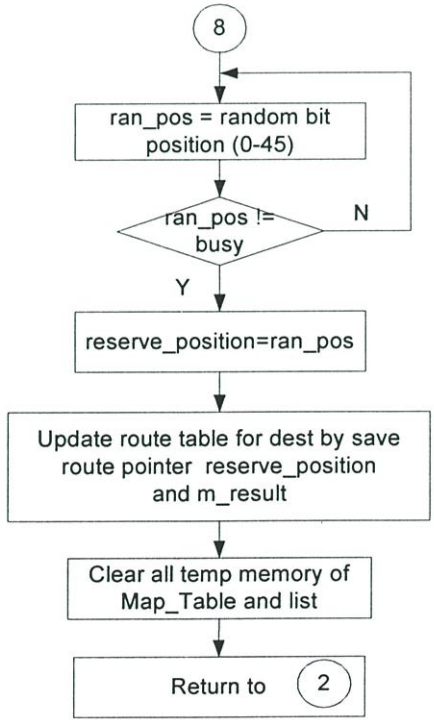
รูปที่ 3.23 โฟล์วชาร์ตฟังก์ชันการค้นหาเส้นทาง (ต่อ)



รูปที่ 3.24 โฟล์วชาร์ตฟังก์ชันการค้นหาเส้นทาง (ต่อ)

จากรูปที่ 3.24 หากผลการ AND มีช่องสัญญาณเวลาที่สามารถใช้ได้ จะทำการเก็บผลการ AND ไว้ในลิสต์ที่อยู่ในตำแหน่งถัดไปในตาราง Map_Table เพื่อเก็บข้อมูลไว้ใช้ในการ AND บิตข้อมูลในรอบถัดไป ซึ่งในตัวอย่างรูปที่ 3.22 จะเป็นการ AND ระหว่าง B_{ij} และ B_{km} โดยใช้ตำแหน่งเยื้อง A_{ik} ซึ่งจะได้ B_{im} ส่วน B_n นั้นไม่มีคู่ในการ AND ก็จะเก็บข้อมูล B_n โดยตรง ดังนั้นข้อมูลที่จะเก็บลงในลิสต์ subm_rte_list ในตำแหน่งที่ 3 ของตาราง Map_Table คือ B_{im} และ B_n ส่วน shift_pos_list จะเก็บค่า A_{in} ซึ่งมีค่าเท่ากับ $A_{ik} + A_{kn}$ และในการเทียบบิตในรอบที่สามนี้จะทำการ AND ระหว่าง B_{im} และ B_n โดยใช้ตำแหน่งเยื้อง A_{in} และเมื่อทำการเทียบบิตครบ 3 รอบก็จะได้ผลการเทียบบิตของข้อมูลสถานะการใช้งานช่องสัญญาณเวลาของโหนดทั้งหมดในเส้นทาง ผลที่ได้จะเป็น B_{in} ซึ่งเป็นผลการเทียบบิตข้อมูลตั้งแต่เราท์เตอร์ i ถึงเราท์เตอร์ n

เมื่อทำการเทียบบิตทั้งเส้นทางแล้ว จะต้องนำมาพิจารณาเลือกตำแหน่งว่างที่ซิงโครนัสที่สามารถจองได้โดยการสุ่มเพื่อเลือกตำแหน่งใน B_{in} ที่มีบิตเป็น 1 ดังรูปที่ 3.25 เมื่อเลือกตำแหน่งได้แล้วก็จะทำการเก็บเส้นทางและตำแหน่งช่องสัญญาณเวลาไว้ในตารางเส้นทางแล้วจึงทำการหาเส้นทางสำหรับโหนดปลายทางอื่นต่อไป ดังโฟล์วชาร์ตในรูปที่ 3.18



รูปที่ 3.25 โฟล์วชาร์ตฟังก์ชันการค้นหาเส้นทาง (ต่อ)

ในส่วนที่เป็นการหาเส้นทางที่สั้นที่สุดนั้นจะเลือกเส้นทางที่มีค่าภาระการใช้งานลิงค์น้อยที่สุด โดยที่ค่าภาระการใช้งานลิงค์นั้นเป็นฟังก์ชันที่คำนวณจากเวลาหน่วงเนื่องจากการส่งผ่านสายสื่อสารและแบนด์วิดท์ที่สามารถใช้งานได้ซึ่งคำนวณจากข้อมูลสถานะการใช้งานช่องสัญญาณเวลาจากนั้นจึงใช้การเทียบบิตเพื่อตรวจสอบคุณภาพของเส้นทางอีกครั้งเพื่อลดอัตราการปฏิเสธการร้องขอเส้นทาง เนื่องจากเส้นทางที่เลือกเป็นเส้นทางที่มีตำแหน่งว่าง ที่ซิงโครนัส ในงานวิจัยนี้กำหนดให้เป็นการหาเส้นทางที่เราท์เตอร์ต้นทางดังนั้นฟังก์ชันการค้นหาเส้นทางจะมีแต่ที่อุปกรณ์เราท์เตอร์จำลองเท่านั้น ส่วนที่เป็นอุปกรณ์สวิตชิ่งโหนดนั้น ในสถานะ route นี้จะทำหน้าที่เพียง

<u>สถานะ call</u>

ที่สถานะนี้จะเป็นกรณีที่มีแพ็กเกตร้องขอการเชื่อมโยงเส้นทางหรือแพ็กเกตตอบรับเข้ามาซึ่งในกระบวนการร้องเส้นทางเชื่อมโยงนั้นจะสัมพันธ์กับการจัดลำดับแพ็กเกตเพื่อส่งออก เนื่องจากการทำการสำรองช่องสัญญาณเวลาจะเป็นการเลือกตำแหน่งที่จะเก็บข้อมูลการแปลส่วนหัวของแพ็กเกตในกระบวนการจัดลำดับแพ็กเกตเพื่อส่งออก ซึ่งวิธีการจัดลำดับแพ็กเกตเพื่อส่งออกในหลักการ W-TDM นั้นนอกจากจะเพิ่มความยืดหยุ่นของช่วงเวลาให้บริการแล้วยังเพิ่มความยืดหยุ่นในการจองช่องสัญญาณเวลาสำหรับการจองช่องสัญญาณเวลาแบบตายตัวในเฟรมขนาดคงที่ด้วย เพราะหากแพ็กเกตเข้ามาแล้วยังอยู่ในช่วงวินโดว์ แพ็กเกตจะยังได้รับการบริการโดยส่งไปเก็บรอในบัฟเฟอร์ ณ ตำแหน่งช่องสัญญาณเวลาที่จองไว้ ทำให้ได้รับการบริการส่งออกไปในตำแหน่งเดิมดังนั้นในการจองช่องสัญญาณเวลาจึงสามารถจองตำแหน่งที่อยู่ถัดจากตำแหน่งซิงโครนัสได้แต่ต้องเป็นตำแหน่งที่อยู่ภายในขอบเขตของวินโดว์ การทำงานที่สถานะ call ของอุปกรณ์เราท์เตอร์จำลองนั้นจะเป็นดังรูปที่ 3.26 เมื่อนำแพ็กเกตออกจากลิสต์แล้วจะต้องเข้าถึงข้อมูลในแพ็กเกตเพื่อพิจารณาว่าเป็นแพ็กเกตประเภทไหน

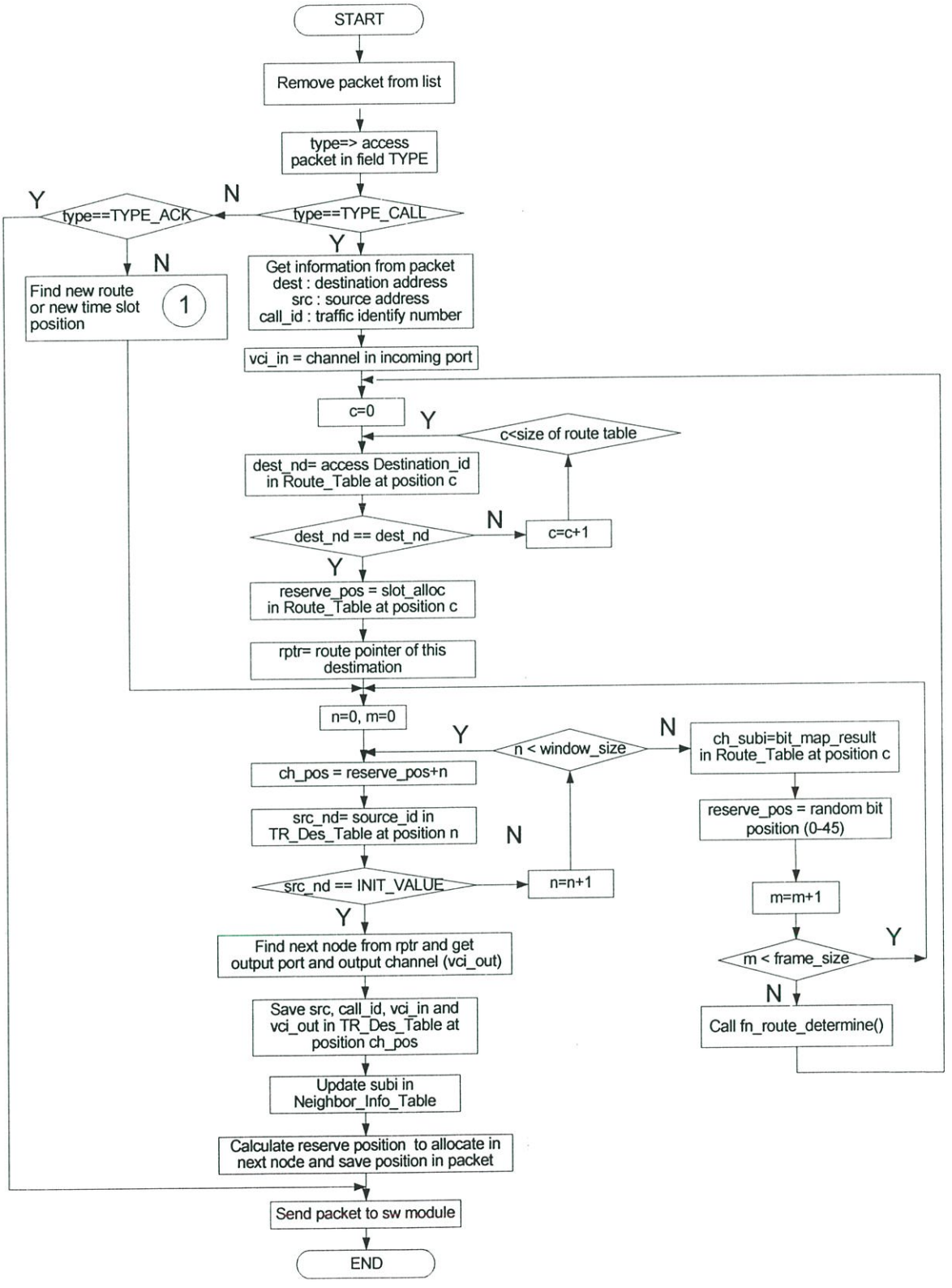
หากเป็นแพ็กเกตร้องขอเพื่อเชื่อมโยงเส้นทาง ข้อมูลที่จะนำมาใช้จะเป็นหมายเลขโหนดต้นทาง หมายเลขปลายทาง เลขลำดับของทราฟฟิกและหมายเลขช่องสื่อสารที่รับแพ็กเกตเข้ามาเริ่มกระบวนการโดยพิจารณาว่าเป็นทราฟฟิกที่ต้องการส่งไปยังโหนดปลายทางใด จากนั้นจะตรวจสอบเส้นทางในตารางเส้นทางและนำค่าตำแหน่งช่องสัญญาณเวลาที่เก็บไว้ในตารางเส้นทางมาเป็นตำแหน่งที่จะจองและจัดเก็บข้อมูลในการจัดการส่วนหัวของแพ็กเกตในกระบวนการจัดลำดับแพ็กเกตเพื่อส่งออกซึ่งต้องตรวจสอบโหนดถัดไปในเส้นทางแล้วดูว่าจะต้องส่งออกทางพอร์ใด และทางช่องสื่อสารขาออกใดจากนั้นจึงนำข้อมูลหมายเลขโหนดต้นทาง หมายเลขลำดับทราฟฟิก ช่องสื่อสารขาเข้าและขาออกมาเก็บในตำแหน่งดังกล่าวในตารางวงจรเสมือนก่อนที่จะเก็บข้อมูลลงในตำแหน่งดังกล่าวต้องตรวจสอบสถานะการใช้งานในขณะนั้นก่อนว่าว่างหรือไม่ ซึ่งสถานะว่างนั้น

จะถูกกำหนดให้เป็นค่า INIT_VALUE ตั้งแต่สถานะเริ่มต้น และหากค่าที่ตรวจสอบไม่ใช่ค่า INIT_VALUE แสดงว่าตำแหน่งนั้นถูกใช้งานแล้วจะต้องทำการพิจารณาตำแหน่งข้างเคียงที่ยังอยู่ในขอบเขตของวินโดว์ว่ามีตำแหน่งว่างหรือไม่ หากไม่มีตำแหน่งว่างในขอบเขตของวินโดว์จะต้องนำชุดข้อมูลที่เป็นผลการเทียบบิตขั้นสุดท้ายมาทำการเลือกสุ่มเพื่อเลือกตำแหน่งว่างที่ซิงโครนัสตำแหน่งอื่นใหม่ซึ่งจะทำการพิจารณาได้สูงสุด 46 ครั้ง เท่าขนาดของเฟรม หากยังไม่มีตำแหน่งที่สามารถใช้งานได้จะต้องเรียกฟังก์ชันการหาเส้นทางมาใช้แล้วทำการหาเส้นทางใหม่และเก็บข้อมูลใหม่นี้ลงในตารางเส้นทางแล้วจึงทำกระบวนการหาตำแหน่งว่างที่จะใช้ในการจองอีกครั้ง และเมื่อจองได้แล้วจะคำนวณหาตำแหน่งที่ควรจะจองในโหนดถัดไปแล้วเก็บค่าตำแหน่งดังกล่าวลงไปในแพ็กเกตเพื่อบอกตำแหน่งที่ต้องการจองต่อไป

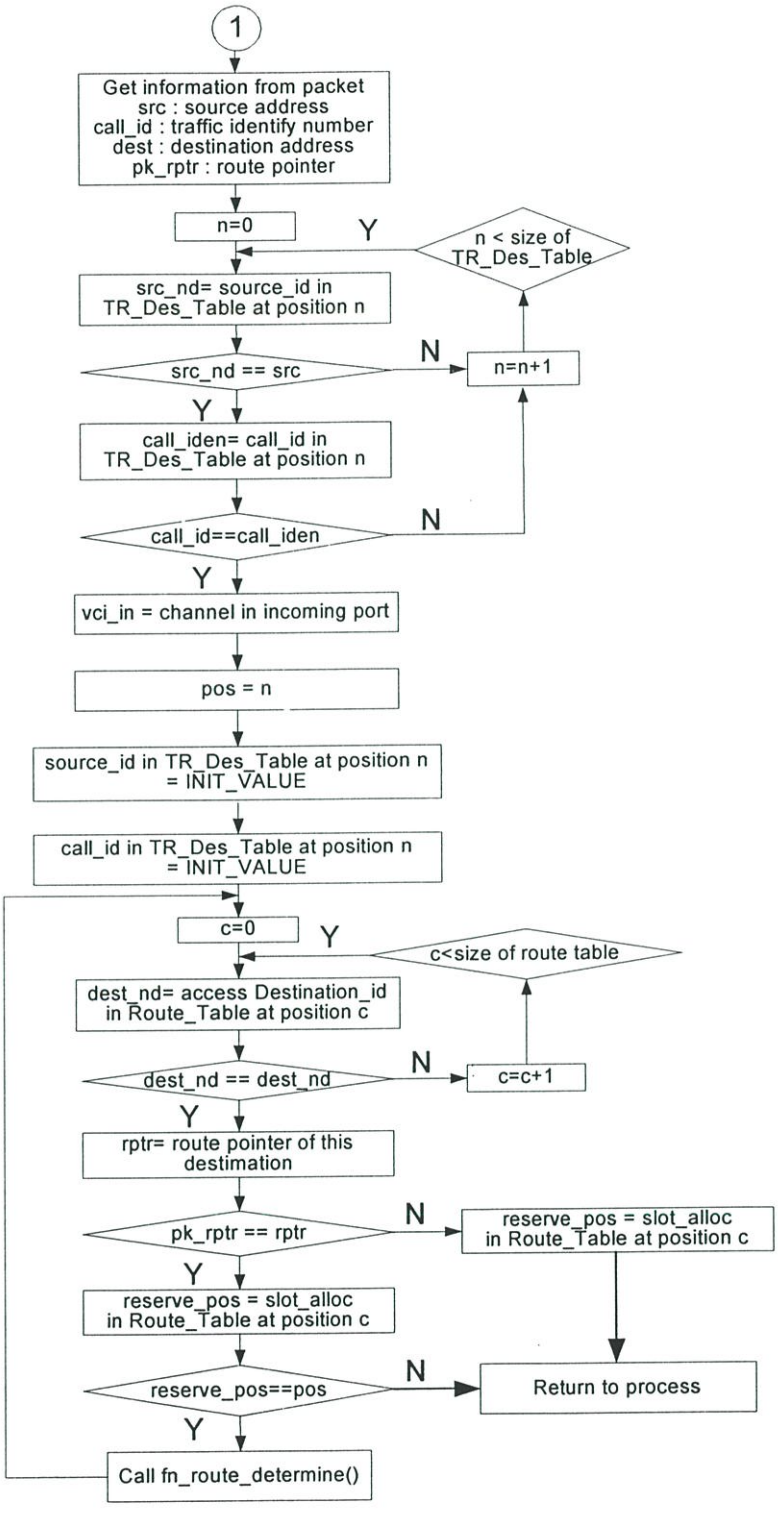
หากแพ็กเกตที่เข้ามาเป็นแพ็กเกตตอบรับแบบตอบรับ เป็นการยืนยันการใช้งานเส้นทางและจองช่องสัญญาณเวลา ซึ่งอาจจะทำการเก็บค่าสถิติหรือค่าอื่น ๆ เกี่ยวกับแพ็กเกตตอบรับนี้ได้ในที่นี้จะส่งแพ็กเกตไปยังมอดูลสวิตช์เลย หากแพ็กเกตเป็นแพ็กเกตตอบรับแบบปฏิเสธ จะต้องทำการคืนตำแหน่งที่จองแล้วหาเส้นทางให้กับทราฟฟิกดังกล่าวใหม่หรือหาตำแหน่งช่องสัญญาณเวลาใหม่ซึ่งจะมีการทำงานดังรูปที่ 3.27

สถานะ call ที่อุปกรณ์สวิตช์นั้นจะมีโฟล์วชาร์ตการทำงานดังรูปที่ 3.28 คล้ายกับสถานะ call ที่เราท์เตอร์แต่จะต่างกันที่ อุปกรณ์สวิตช์จำลองนี้จะรับค่าตำแหน่งช่องสัญญาณเวลาจากแพ็กเกตซึ่งมีการคำนวณทุกครั้งหลังจากที่ทำการจองช่องสัญญาณเวลาได้แล้ว จะพิจารณาเวลาหน่วงและตำแหน่งเยื้องที่ทำให้ซิงโครนัสในโหนดถัดไป แล้วนำมาบวกกับตำแหน่งที่ทำการจองช่องสัญญาณเวลาทำให้ได้ตำแหน่งที่เหมาะสมในการจองของโหนดถัดไปดังนั้นในอุปกรณ์สวิตช์จึงทำการเข้าถึงข้อมูลในแพ็กเกตแล้วนำค่าตำแหน่งดังกล่าวนี้มาเป็นตำแหน่งที่จะใช้ในการจองและหากตำแหน่งดังกล่าวนี้ไม่ว่างก็จะพิจารณาตำแหน่งข้างเคียงที่อยู่ในช่วงวินโดว์หากไม่มีตำแหน่งว่างภายในวินโดว์จะต้องทำการแจ้งการปฏิเสธการร้องขอเส้นทาง

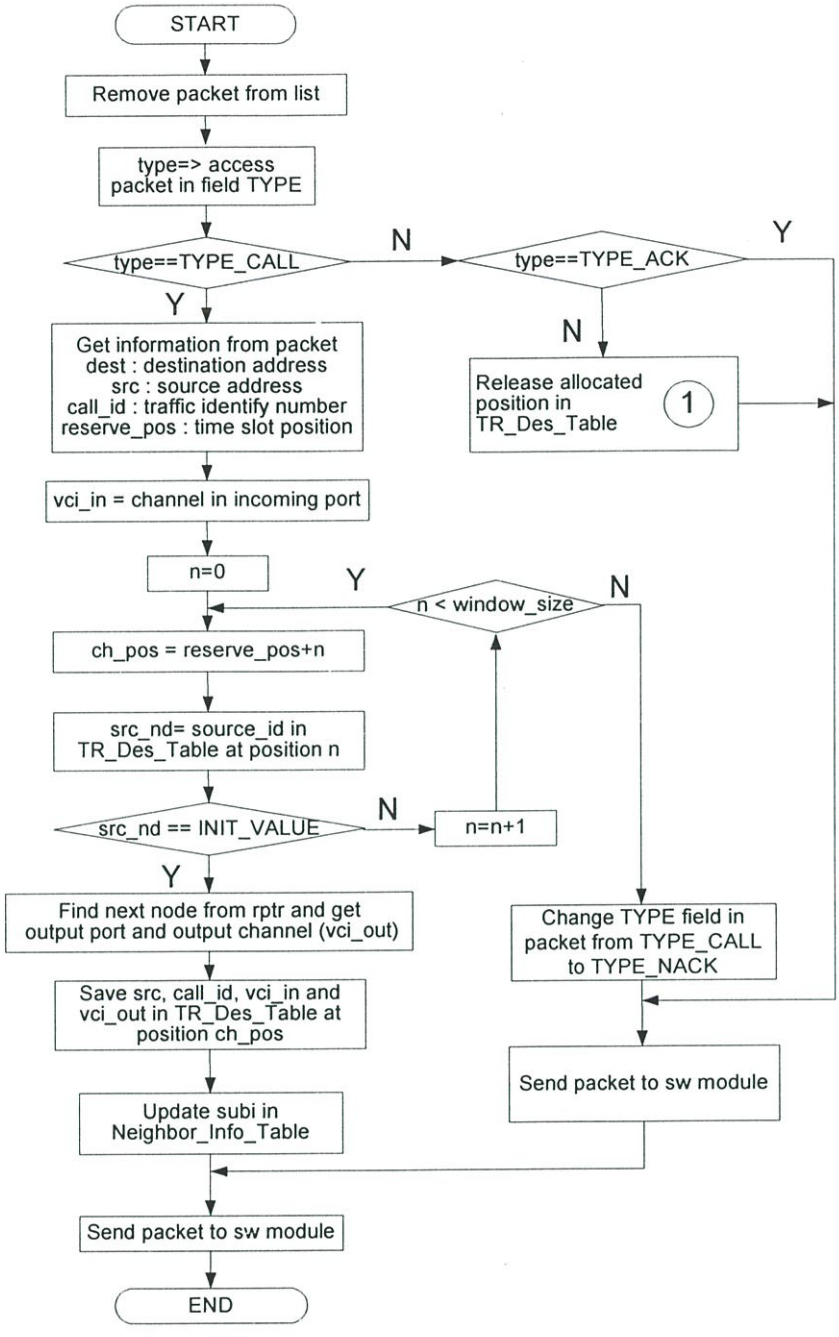
การแจ้งปฏิเสธในการจำลองนี้ได้ใช้แพ็กเกตร้องขอเพื่อเชื่อมโยงเส้นทางแต่เปลี่ยนชนิดของแพ็กเกตเป็นแพ็กเกตตอบรับแบบปฏิเสธเนื่องจากในแพ็กเกตนี้มีข้อมูลของแพ็กเกตร้องขอการเชื่อมโยงเส้นทางอยู่ทำให้สามารถตรวจสอบได้ง่ายว่าเป็นการปฏิเสธของทราฟฟิกใดและเมื่อส่งลงไปยังสวิตช์ก็สามารถตรวจสอบได้ว่าจะส่งแพ็กเกตตอบรับแบบปฏิเสธย้อนกลับไปทางใด ทั้งนี้เพื่อทำการยกเลิกการจองช่องสัญญาณในเส้นทาง จนกระทั่งถูกส่งกลับไปถึงเราเตอร์ต้นทางที่ส่งแพ็กเกตร้องขอการเชื่อมโยงเส้นทางมา เพื่อให้เราท์เตอร์ต้นทางทำการหาเส้นทางใหม่หรือคำนวณตำแหน่งช่องสัญญาณเวลาใหม่ ซึ่งการส่งแพ็กเกตตอบรับแบบปฏิเสธจะมีการทำงานดังโฟล์วชาร์ตในรูปที่ 3.29



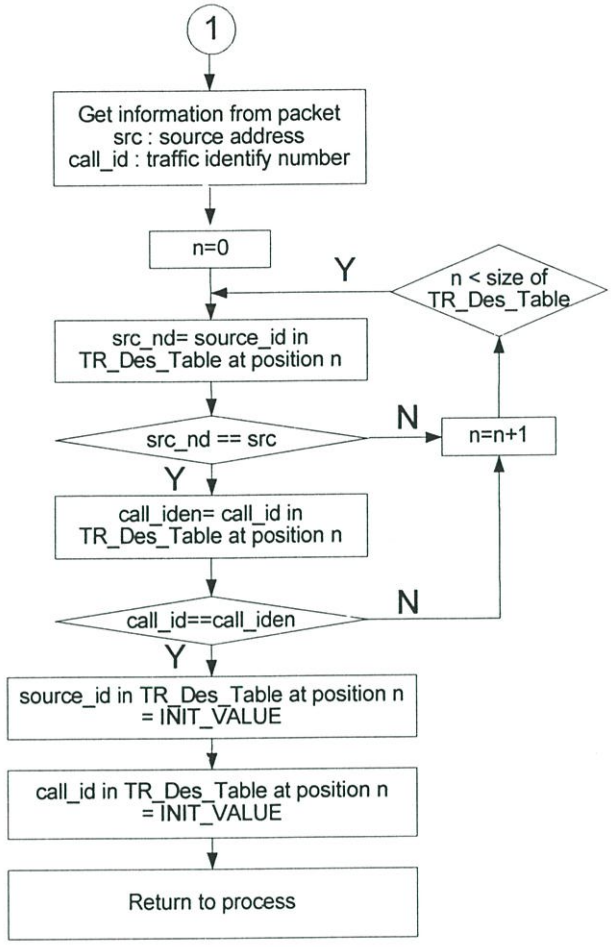
รูปที่ 3.26 โฟล์วชาร์ตการทำงานของสถานะ call ในอุปกรณ์เราท์เตอร์จำลอง



รูปที่ 3.27 โฟล์วชาร์ตการทำงานของสถานะ call ในอุปกรณ์เราท์เตอร์จำลอง (ต่อ)



รูปที่ 3.28 โฟล์วชาร์ตการทำงานของสถานะ call ในอุปกรณ์สวิตช์จำลอง



รูปที่ 3.29 โฟล์วชาร์ตการทำงานของจำลอง กรณีที่ได้รับแพ็กเกตตอบรับแบบปฏิเสธ

3.2.2.3 **มอดูล** sw

การทำงานในมอดูล sw มีแผนผังสถานะดังรูปที่ 3.30 ซึ่งจะแยกแพ็กเกตเก็บลงลิสต์ตามประเภทของแพ็กเกตและถูกดำเนินการตามชนิดของแพ็กเกต กรณีที่เป็นแพ็กเกตแจ้งสถานะการใช้งานเส้นทางเชื่อมโยง จะถูกส่งกระจายออกไปยังโหนดอื่น ๆ โดยถูกส่งออกไปทุกทิศทางยกเว้นทางพอร์ตที่ได้รับเข้ามา ส่วนแพ็กเกตที่เป็น signaling อื่น ๆ จะถูกส่งไปตามเส้นทางที่ระบุในแพ็กเกต แล้วหาแอดเดรสของโหนดถัดไปจากเส้นทางที่แนบมากับแพ็กเกตนั้น ส่วนแพ็กเกตข้อมูลนั้นจะต้องทำกระบวนการจัดลำดับแพ็กเกตเพื่อส่งออก ตามหลักการของ W-TDM และรอในบัฟเฟอร์ขาออกในตำแหน่งที่สัมพันธ์กันเพื่อรักษาการซิงโครนัส กระบวนการในมอดูลสวิตช์ของอุปกรณ์เราท์เตอร์จำลองและอุปกรณ์สวิตช์จำลองจะมีส่วนที่ต่างกันซึ่งจะได้อธิบายเป็นโฟล์วชาร์ตการทำงานดังต่อไปนี้

สถานะเริ่มต้น (init state)

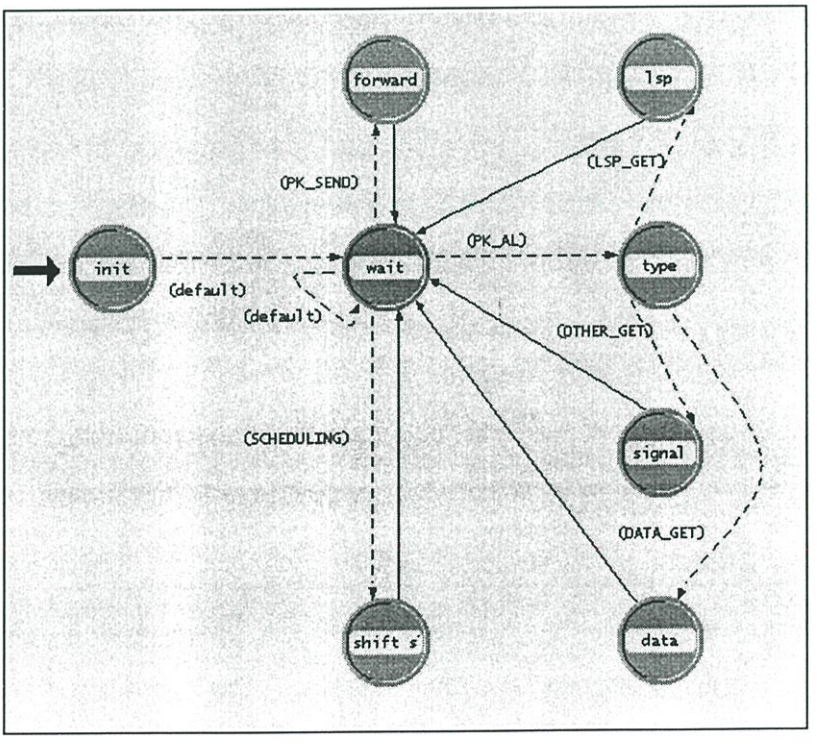
จะทำการเซตค่าเริ่มต้นต่าง ๆ รวมทั้งสร้างและกำหนดหน่วยความจำให้กับลิสต์และตารางต่าง ๆ ที่จะต้องใช้ในมอดูลนี้ มีการติดตั้งและเก็บข้อมูลการเชื่อมโยงภายในระบบเครือข่ายจำลองและมีการกำหนดตัวชี้เพื่อเชื่อมโยงข้อมูลในตารางวงจรเสมือน จากนั้นจะไปรอที่สถานะ wait เพื่อรอรับอินเตอร์รัพท์ต่าง ๆ

สถานะรอคอย (wait state)

รอรับอินเตอร์รัพท์ต่าง ๆ ซึ่งในมอดูลสวิตช์นี้จะมีการอินเตอร์รัพท์ของเหตุการณ์เมื่อได้รับแพ็กเกตเข้ามา ซึ่งจะถูกส่งไปแยกสถานะของแพ็กเกตที่สถานะแยกประเภทของแพ็กเกต_และอินเตอร์รัพท์โดยเวลา เพื่อทำการเลื่อนตัวชี้ในตารางวงจรเสมือน และตัวชี้ที่บัฟเฟอร์ขาออก ซึ่งมีความสัมพันธ์กัน

สถานะแยกประเภทของแพ็กเกต (type state)

เมื่อมีแพ็กเกตเข้ามาจะทำการเข้าถึงข้อมูลในแพ็กเกตเพื่อตรวจสอบว่าเป็นแพ็กเกตประเภทใดจากนั้นจะทำการส่งแพ็กเกตไปทำกระบวนการในสถานะอื่นที่สัมพันธ์กับประเภทของแพ็กเกตต่อไป โดยจะเก็บแพ็กเกตลงในลิสต์ของสถานะนั้น



รูปที่ 3.30 โฟล์วชาร์ตสถานะภายในมอดูลสวิตช์

สถานะจัดการแพ็กเกตแจ้งสถานะการใช้งานเส้นทางเชื่อมโยง (lsp state)

จะนำแพ็กเกตออกจากลิสต์แล้วพิจารณาว่าแพ็กเกตนี้ถูกรับเข้ามาทางพอร์ตใดจากนั้นจึงส่งแพ็กเกตกระจายออกไปทางพอร์ตอื่น ๆ ที่จะไปยังโหนดข้างเคียงที่เป็นสวิตช์หรือเราท์เตอร์

สถานะจัดการแพ็กเกตควบคุม (signal state)

ในสถานะนี้จะพิจารณาแพ็กเกตร้องขอเส้นทางและแพ็กเกตตอบรับ ซึ่งจะมีทิศทางการส่งแบบทิศทางไปข้างหน้าและทิศทางย้อนกลับ หากเป็นแพ็กเกตร้องขอการเชื่อมโยงจะทำการเข้าถึงข้อมูลในเส้นทางที่แนบมากับแพ็กเกตแล้วหาค่าโหนดถัดไป เมื่อทราบหมายเลขโหนดถัดไปจะทราบหมายเลขเส้นทางเชื่อมโยงหมายเลขพอร์ตและหมายเลขช่องสื่อสารที่จะส่งแพ็กเกตออกไป

ส่วนแพ็กเกตตอบรับนั้น หากโหนดปัจจุบันเป็นสวิตช์โหนดจะทำการพิจารณาโหนดที่อยู่ในตำแหน่งถัดไปในทิศทางย้อนกลับแล้วส่งออกไปยังพอร์ตที่จะไปยังโหนดนั้นทั้งกรณีที่เป็นแบบตอบรับและปฏิเสธ หากโหนดปัจจุบันเป็นเราท์เตอร์โหนดจะพิจารณาแยกกรณี หากเป็นแพ็กเกตแบบตอบรับต้องพิจารณาก่อนว่าหมายเลขโหนดแรกสุดในเส้นทางมีค่าเท่ากับหมายเลขของเราท์เตอร์โหนดหรือไม่ ถ้าเท่าแสดงว่าเราท์เตอร์ปัจจุบันเป็นเราท์เตอร์ต้นทางซึ่งเชื่อมต่ออยู่กับโหนดต้นทางที่เป็นโหนดข้างเคียง ดังนั้นจะพิจารณาหมายเลขโหนดต้นทางในแพ็กเกตแล้วส่งออกไปทางพอร์ตซึ่งจะไปยังโหนดต้นทางนั้น กรณีที่หมายเลขโหนดแรกสุดในเส้นทางมีค่าไม่เท่ากับหมายเลขของเราท์เตอร์โหนดปัจจุบัน แสดงว่าเราท์เตอร์ปัจจุบันเป็นเราท์เตอร์ปลายทางดังนั้นจะพิจารณาโหนดที่อยู่ในตำแหน่งถัดไปในทิศทางย้อนกลับแล้วส่งออกไปยังพอร์ตที่จะไปยังโหนดนั้น ส่วนกรณีที่เป็นแพ็กเกตตอบรับแบบปฏิเสธนั้นหากเป็นเราท์เตอร์ต้นทางจะทำการลบแพ็กเกตทิ้งที่มอดูล mgmt แล้วสร้างแพ็กเกตร้องขอเพื่อเชื่อมโยงเส้นทางออกมาแทน ดังนั้นกรณีที่มอดูล sw ที่เราท์เตอร์ได้รับแพ็กเกตตอบรับแบบปฏิเสธจึงเป็นกรณีที่เป็นการปฏิเสธจากโหนดเราท์เตอร์ปลายทางเอง ซึ่งในมอดูล sw จะพิจารณาโหนดที่อยู่ในตำแหน่งถัดไปในทิศทางย้อนกลับแล้วส่งออกไปยังพอร์ตที่จะไปยังโหนดนั้น

สถานะเลื่อนลำดับตัวชี้ในการจัดลำดับเพื่อส่งออก (shift state)

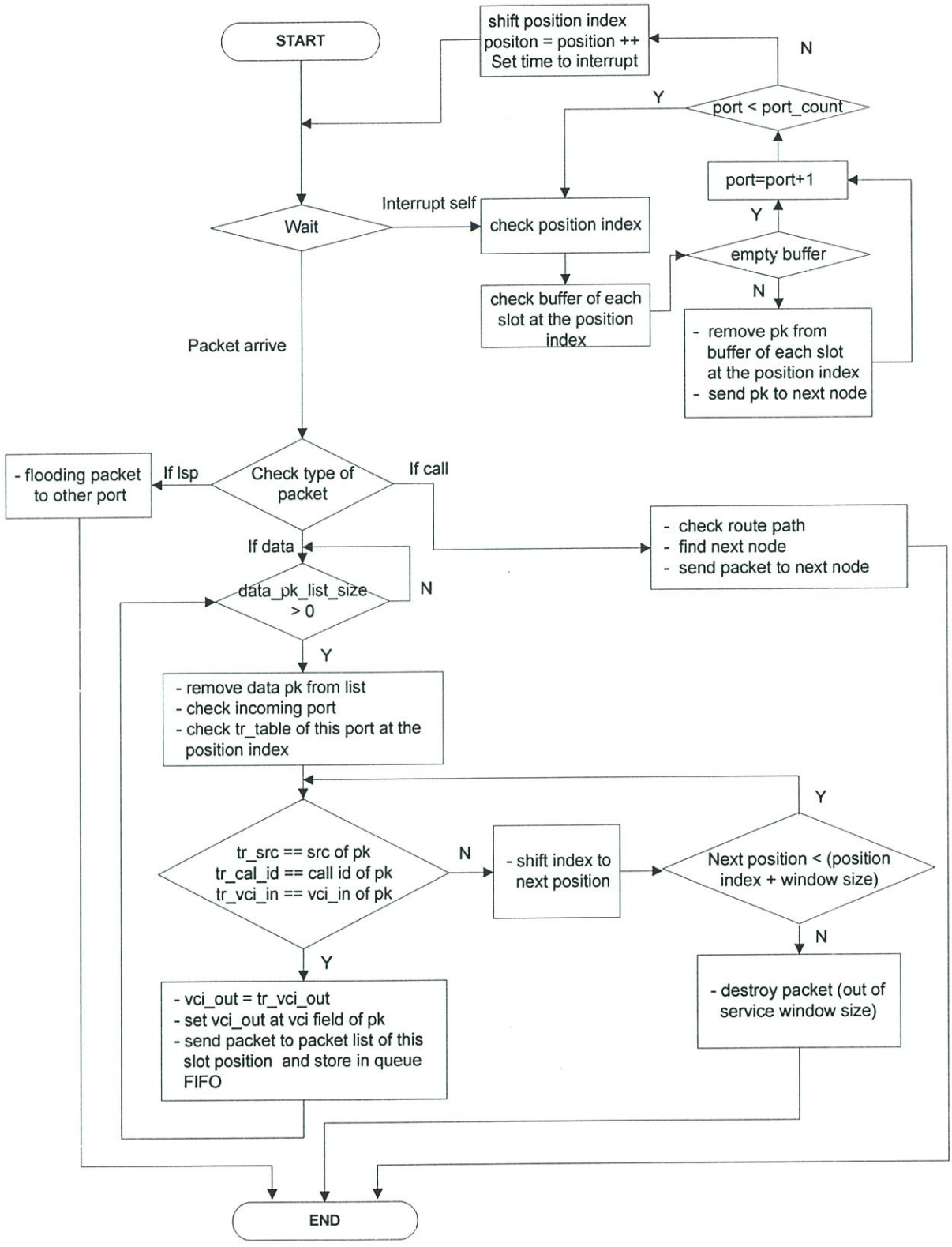
ในสถานะนี้จะมีการตั้งเวลาอินเตอร์รัพท์เพื่อเลื่อนลำดับตัวชี้เวลาที่ใช้ในการอินเตอร์รัพท์จะถูกเซตไว้ให้เท่ากับเวลา 1 ช่องสัญญาณเวลาเพื่อเป็นเวลาในการให้บริการแพ็กเกตในกระบวนการจัดลำดับแพ็กเกตเพื่อส่งออกซึ่งอยู่ในสถานะ data และตัวชี้นี้จะสัมพันธ์กับตัวชี้ตำแหน่งในบัฟเฟอร์ขาออกซึ่งอยู่ในสถานะ forward ดังนั้นการให้บริการในกระบวนการจัดลำดับแพ็กเกตเพื่อส่งออกและกระบวนการส่งแพ็กเกตออกไปยังโหนดถัดไปจึงทำงานสอดคล้องกัน

สถานะจัดการแพ็กเกตข้อมูล (data state) และสถานะส่งแพ็กเกต (forward state)

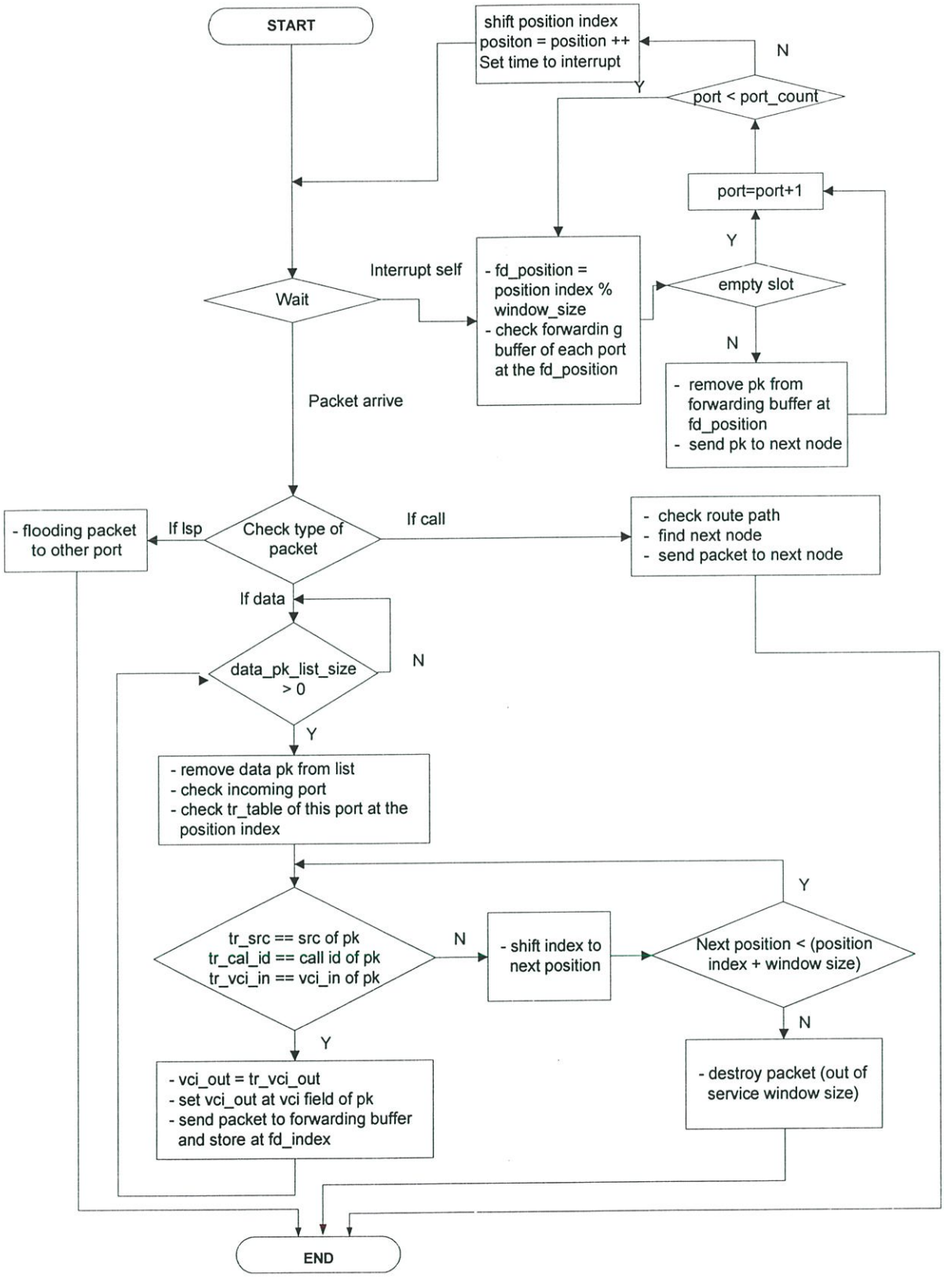
สถานะทั้งสองนี้ทำงานสอดคล้องกันจึงจะทำการอธิบายพร้อมกันดังจะเห็นในโฟล์วชาร์ตการทำงานในรูปที่ 3.31 และ 3.32 ซึ่งหากแพ็กเกตมีการรับส่งอย่างซิงโครนัสตามเวลาที่จองไว้จะทำให้แพ็กเก็ตได้รับการบริการทันทีที่เข้ามาถึง ที่เราท์เตอร์นี้ ขนาดของวินโดว์จะถูกเซตให้เท่ากับขนาดของเฟรมเพื่อเพิ่มช่วงการพิจารณาในการจองช่องสัญญาณเวลาช่องสัญญาณเวลาที่สามารถจองได้ให้มากขึ้น ทำให้ลดการปฏิเสธการร้องขอเส้นทางเชื่อมโยงตั้งแต่โหนดแรก ดังนั้นที่เราท์เตอร์ต้นทางจึงสามารถเลือกจองตำแหน่งว่างที่ซิงโครนัสตำแหน่งใดก็ได้ในเฟรม การพัฒนาหลักการ W-TDM นั้น จะกระทำที่สถานะ data และสถานะ forward โดยที่สถานะ data นั้นจะรับแพ็กเกตเข้ามาแล้วพิจารณาตัวชี้ตำแหน่งในตารางวงจรเสมือน ณ ขณะนั้นและทำการจัดการข้อมูลส่วนหัวของแพ็กเกตโดยเปรียบเทียบค่า VCI ของแพ็กเกต กับ VCI_IN ในตารางวงจรเสมือน ณ ตำแหน่งที่ชี้อยู่ในปัจจุบัน เมื่อค่า VCI ของแพ็กเกตตรงกับค่า VCI_IN ในตารางวงจรเสมือน ก็จะนำค่า VCI_OUT มาเซตลงไปในแพ็กเกตแทน ซึ่งค่า VCI_OUT นี้จะถูกนำไปใช้ในกระบวนการจัดลำดับแพ็กเกตเพื่อส่งออกในโหนดถัดไป แต่หากค่า VCI ณ ตำแหน่งที่ชี้อยู่ในปัจจุบันมีค่าไม่ตรงกัน จะต้องทำการเข้าถึงที่ตำแหน่งอื่นในตารางวงจรเสมือน โดยเลื่อนลำดับการตรวจสอบไปยังตำแหน่งถัดไปเรื่อย ๆ ซึ่งขอบเขตในการพิจารณานี้สามารถเลื่อนไปได้เท่ากับขนาดของวินโดว์ ซึ่งขนาดของวินโดว์นี้จะระบุเป็นจำนวนช่องสัญญาณเวลาที่อยู่ในขอบเขตของการพิจารณา โดยเริ่มนับตั้งแต่ตำแหน่งของตัวชี้ปัจจุบันเป็นต้นไป

การจำลองในงานวิจัยนี้กำหนดให้ขนาดวินโดว์ที่เราท์เตอร์มีขนาดเท่ากับขนาดของเฟรม ดังนั้นหากทราฟฟิกนี้มีการจองช่องสัญญาณเวลาไว้ให้ไม่ว่าตำแหน่งใดในเฟรม แพ็กเกตของทราฟฟิกนี้จะสามารถส่งออกไปเก็บยังบัฟเฟอร์ขาออกที่ตำแหน่งที่จองไว้ได้และรอเพื่อนำส่งไปยังสวิตชิ่งโหนดถัดไป การใช้บัฟเฟอร์สำหรับทุกช่องสัญญาณเวลาร่วมกับการเลื่อนลำดับตัวชี้ตำแหน่งให้บริการในช่วงเวลาที่คงที่จะเป็นการบังคับอัตราการส่งแพ็กเกตให้คงที่ ซึ่งสามารถช่วยลดภาระในส่วนนี้ที่โหนดสื่อสารปลายทางได้ และเพิ่มประสิทธิภาพของการบังคับอัตราเร็วในการสื่อสารที่มีการจองช่องสัญญาณเวลาตายตัวในเฟรมคงที่ได้ดีขึ้น

ส่วนมอดูล sw ของอุปกรณ์สวิตช์จำลองนั้นไม่มีบัฟเฟอร์สำหรับทุกช่องสัญญาณเวลาแต่จะมีบัฟเฟอร์สำหรับในแต่ละพอร์ตซึ่งมีขนาดเท่ากับขนาดของวินโดว์ โดยจะกำหนดขอบเขตของวินโดว์ให้เล็กกว่าเฟรม ซึ่งในการทดลองในงานวิจัยนี้กำหนดให้มีขนาดเท่ากับ 10 ทั้งนี้เพื่อให้เป็นไปตามคุณสมบัติของสวิตช์ในหลักการ W-TDM ซึ่งหากวินโดว์มีขนาดใหญ่แม้จะยืดหยุ่นในการจองช่องสัญญาณเวลาแต่ก็จะทำให้เวลาหน่วงในกระบวนการจัดลำดับแพ็กเกตเพื่อส่งออกมีเวลามากกว่าวินโดว์ขนาดเล็ก



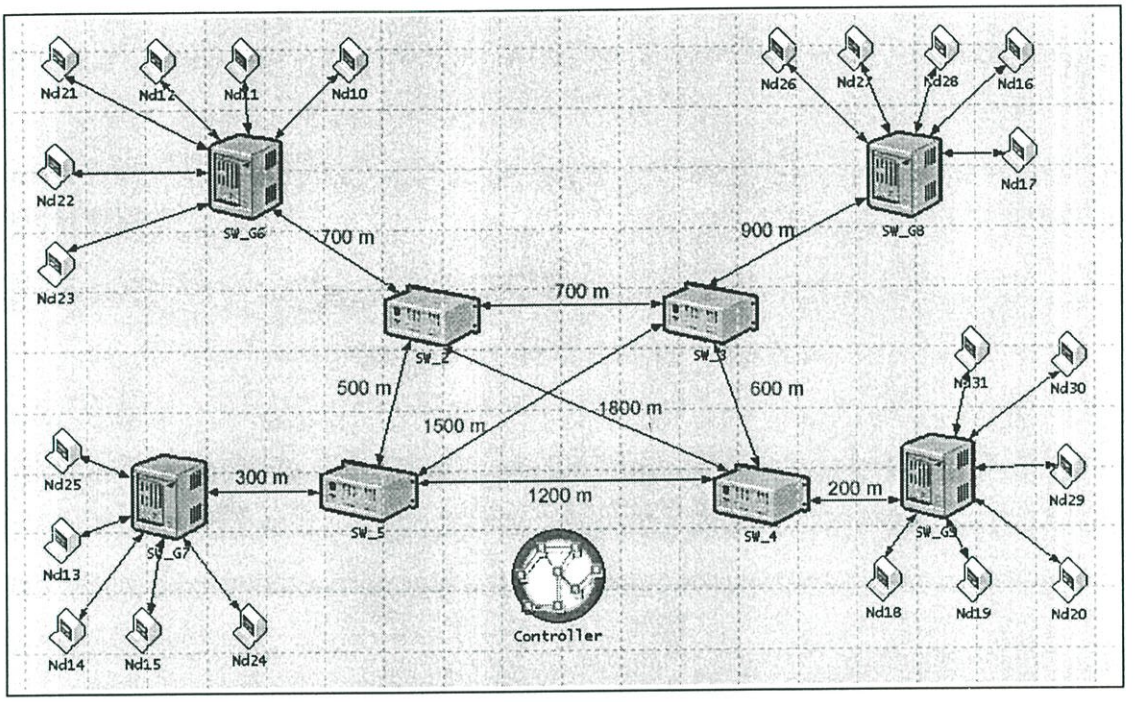
รูปที่ 3.31 โฟล์วชาร์ตการทำงานในมอดูลสวิตช์ที่อุปกรณ์เราท์เตอร์จำลอง



รูปที่ 3.32 โฟล์วชาร์ตการทำงานในมอดูลสวิตช์ที่อุปกรณ์สวิตช์จำลอง

บทที่ 4
การจำลองและผลการจำลอง

4.1 กระบวนการจำลองระบบเครือข่าย



รูปที่ 4.1 ระบบเครือข่ายจำลองที่ใช้ในการทดลอง

4.1.1 ข้อกำหนดของอุปกรณ์เครือข่ายจำลอง

● อุปกรณ์สื่อสารปลายทางประกอบด้วยโหนด Nd10, Nd11, Nd12, Nd13, Nd14, Nd15, Nd16, Nd17, Nd18, Nd19, Nd20, Nd21, Nd22, d23, Nd24, Nd25 Nd26, Nd27, Nd28, Nd29, Nd30 และ Nd31

- การกำหนดเวลาเริ่มต้นที่ต้องการสร้างแพ็กเกตร้องขอเพื่อเชื่อมโยงเส้นทางนั้นจะใช้การสุ่มค่าทางเวลาตั้งแต่ 0 ถึง 2 วินาที และใช้คาบเวลาในการสร้างแพ็กเกตร้องขอเพื่อเชื่อมโยงเส้นทางของทราฟฟิกถัดไปโดยใช้การสุ่มค่าทางเวลาตั้งแต่ 0 ถึง 2 วินาที แล้วจึงนำไปรวมกับเวลาของการสร้างแพ็กเกตก่อนหน้า

- สุ่มค่าเพื่อเลือกฟังก์ชันการกระจายซึ่งจะมีอยู่ 2 กรณีคือ การกระจายแบบคงที่ (constant function) และแบบเอกซ์โปเนนเชียล (exponential function)

- สุ่มค่าหมายเลขโหนดปลายทางจากหมายเลขอุปกรณ์สื่อสารปลายทางจำลองทั้งหมดในระบบเครือข่าย

- สุ่มค่าทางเวลาตั้งแต่ 10 ถึง 15 วินาทีเพื่อหาเวลาที่ต้องการหยุดส่งแพ็กเกตข้อมูลของแต่ละทราฟฟิกเพื่อทำการปลดปล่อยเส้นทางเชื่อมโยงและคืนทรัพยากรแก่ระบบ

● อุปกรณ์สวิตช์ SW_3, SW_4 และ SW_5 และอุปกรณ์เราท์เตอร์ SW_G6, SW_G7, SW_G8 และ SW_G9

- มีการกำหนดจำนวนช่องสัญญาณเวลาในเฟรมเท่ากับ 46 ช่องสัญญาณและกำหนดขนาดของวินโดว์เท่ากับ 10 ช่องสัญญาณเวลา

- ค่าสถานะการใช้งานช่องสัญญาณเวลาในเฟรมของพอร์ตขาออกทุกพอร์ตเป็น 1 ทั้ง 46 ช่องสัญญาณเวลา

- สุ่มค่าทางเวลาตั้งแต่ 0 - 2 วินาทีเพื่อเริ่มส่งแพ็กเกตแจ้งสถานะการใช้งานเส้นทางเชื่อมโยงและคาบเวลาของการส่งแพ้กเกตจะกำหนดไว้ที่ 10 วินาที

4.1.2 โครงสร้างของระบบเครือข่ายจำลอง

ระบบเครือข่ายจำลองที่ใช้ในงานวิจัยครั้งนี้เป็นดังรูปที่ 4.1 โดยระยะห่างระหว่างอุปกรณ์เราท์เตอร์และสวิตช์ในระบบเป็นดังที่แสดงค่าในรูปที่ 4.1 มีหน่วยเป็นเมตร โครงสร้างระบบเครือข่ายจะมีการเชื่อมโยงระหว่างอุปกรณ์เราท์เตอร์และโหนดสื่อสารปลายทางดังนี้ โหนด Nd10, Nd11, Nd12, Nd21, Nd22 และ Nd23 เชื่อมโยงอยู่กับ เราท์เตอร์ SW_G6 โหนด Nd13, Nd14, Nd15, Nd24 และ Nd25 เชื่อมโยงอยู่กับเราท์เตอร์ SW_G7 โหนด Nd16, Nd17, Nd26, Nd27 และ Nd28 เชื่อมโยงอยู่กับเราท์เตอร์ SW_G8 และ โหนด Nd18, Nd19, Nd20, Nd29, Nd30 และ Nd31 เชื่อมโยงอยู่กับเราท์เตอร์ SW_G9 เส้นทางเชื่อมโยงทั้งหมดมีความเร็วในการส่งผ่านสื่อเท่ากับ $3x10^8$ เมตรต่อวินาที และมีอัตราการส่งผ่านข้อมูลเท่ากับ 155 เมกะบิตต่อวินาที ส่วนโหนดควบคุม (Controller) เป็นโหนดที่จะทราบข้อมูลการเปลี่ยนแปลงของระบบอยู่ตลอดเวลาและจะทำการแจ้งข้อมูลที่จำเป็นให้เราท์เตอร์และสวิตช์โหนดเช่นกรณีที่มีการเพิ่มขนาดของระบบ กรณีที่มีการเสียหายของเส้นทางเชื่อมโยงเป็นต้น

4.1.3 การจำลอง

โครงสร้างของระบบเครือข่ายจำลองนี้จะถูกใช้ในการจำลองผล 2 รอบ โดยรอบแรกใช้การคำนวณหาเส้นทางแบบไม่มีวิธีการ SUBM และรอบหลังจะเป็นวิธีการหาเส้นทางที่ใช้อัลกอริทึม SUBM ทั้งนี้เพื่อต้องการเปรียบเทียบประสิทธิภาพของวิธีการหาเส้นทางที่สั้นที่สุดกับวิธีการหาเส้นทางที่ใช้การเทียบบิตจากชุดบิตข้อมูลสถานะการใช้งานช่องสัญญาณเวลาของโหนดในเส้นทาง การจำลองจะเริ่มโดยเราท์เตอร์และสวิตชิ่งโหนดทุกตัวทำการสุ่มเวลาในการเริ่มส่งแพ็กเกตแจ้งสถานะการใช้งาน

เส้นทางเชื่อมโยง ดังนั้นเวลาในการส่งแพ็กเกตแจ้งสถานะการใช้งานเส้นทางเชื่อมโยงจะไม่ตรงกัน ทั้งนี้เพื่อลดปัญหาการวนลูปของการคำนวนหาเส้นทาง ได้กำหนดคาบเวลาของการส่งแพ็กเกตแจ้งสถานะการใช้งานเส้นทางเชื่อมโยงไว้ที่ 10 วินาที ซึ่งทุก ๆ เวลา 10 วินาทีทุกโหนดจะต้องทำการตรวจสอบตัวแปรทราฟฟิก ค่าภาระการใช้ลิงค์ สถานะการใช้งานช่องสัญญาณเวลา และค่าตัวแปร อื่น ๆ ที่ต้องส่งออกไปพร้อมกับแพ็กเกตแจ้งสถานะการใช้งานเส้นทางเชื่อมโยง เพื่อแจ้งข้อมูลสถานะของระบบแก่เราท์เตอร์โหนดและสวิตช์โหนดในระบบ ซึ่งข้อมูลนี้จะถูกนำไปใช้ในกระบวนการค้นหาเส้นทาง และกระบวนการร้องขอเส้นทางต่อไป

การทำงานของโหนดสื่อสารปลายทางแต่ละโหนดเป็นอิสระต่อกัน ตัวแปรของแต่ละทราฟฟิกจะต่างกันไป เช่นในการสร้างแพ็กเกตเพื่อร้องขอการเชื่อมโยงเส้นทาง การติดตั้งค่าโหนดปลายทางจะใช้การสุ่มหมายเลขโหนดปลายทางทางทุกครั้ง เวลาที่ใช้ในการสร้างและส่งแพ็กเกตร้องขอการใช้งานเส้นทางเชื่อมโยงจะเป็นเวลาที่ได้จากการสุ่ม รูปแบบการกระจายในการสร้างและส่งแพ็กเกตข้อมูลของแต่ละทราฟฟิกจะเป็นการเลือกแบบสุ่มระหว่างการกระจายแบบอัตราเร็วคงที่ และการกระจายแบบฟังก์ชันเอกซ์โปเนนเชียล การกำหนดระยะเวลาของการสร้างแพ็กเกตข้อมูลและเวลาในการหยุดสร้างแพ็กเกตเพื่อแจ้งคืนทรัพยากรแก่ระบบก็จะใช้การสุ่มค่าทางเวลาเช่นเดียวกัน ดังนั้นการทำงานของโหนดสื่อสารปลายทางจึงเสมือนเป็นการจำลองอุปกรณ์จริงที่ไม่มีการกำหนดค่าแบบตายตัว ซึ่งจะใกล้เคียงระบบจริงได้ในระดับหนึ่งเนื่องจากทราฟฟิกที่เราสนใจเป็นเฉพาะแบบที่มีอัตราเร็วคงที่ ส่วนทราฟฟิกอื่น ๆ ในระบบเรากำหนดให้อัตราการไหลเป็นแบบเอกซ์โปเนนเชียล เนื่องจากมีการแกว่งและการกระจาย จึงนำมาแทนทราฟฟิกที่ถูกส่งออกมาแบบไม่คงที่แล้วใช้คุณสมบัติของบัฟเฟอร์ขาออกที่เราท์เตอร์ต้นทางในการบังคับอัตราเร็วในการส่งแพ็กเกตให้คงที่อีกที

การกำหนดค่าต่าง ๆ ในอุปกรณ์เราท์เตอร์และสวิตช์ มีสิ่งที่เหมือนกันคือ ขนาดของเฟรมเท่ากับ 46 ช่องสัญญาณเวลาคำนวณได้จากสมการที่ 3.1 กำหนดให้ 1 ช่องสัญญาณเวลาให้บริการส่งผ่านข้อมูล 1 เซลล์ ATM และเวลาในการเลื่อนตัวชี้ตำแหน่งในตารางข้อมูลการจัดลำดับแพ็กเกตเพื่อส่งออกและตัวชี้ตำแหน่งในบัฟเฟอร์ขาออกเท่ากับ 2.735×10^{6} วินาที หาได้จากอัตราส่วนของขนาดแพ็กเกตต่อความเร็วในการส่งผ่านข้อมูล ขนาดของวินโดว์ที่เราท์เตอร์เท่ากับ 46 ส่วนขนาดของวินโดว์ที่สวิตช์เท่ากับ 10 การกำหนดขนาดของวินโดว์ที่เราท์เตอร์เป็น 46 เนื่องจากต้องการเพิ่มความยืดหยุ่นในการเลือกจองช่องสัญญาณเวลา และในการให้บริการในกระบวนการจัดลำดับแพ็กเกตเพื่อส่งออกนั้นสามารถจะเข้าถึงข้อมูลของทุกตำแหน่งในตารางข้อมูลการจัดลำดับแพ็กเกตได้ ทำให้แพ็กเกตของทุกทราฟฟิกที่มีการสำรองช่องสัญญาณเวลาได้รับการบริการและส่งไปเก็บในบัฟเฟอร์ขาออก

เวลาที่กำหนดให้เป็นค่าตายตัวในการจำลองครั้งนี้มีสองค่าคือ เวลาที่ใช้ในการจำลองทั้งหมดเป็น 120 วินาที และเวลาหยุดสร้างแพ็กเกตร้องขอเพื่อเชื่อมโยงเส้นทางของแต่ละโหนดกำหนดเป็น 80

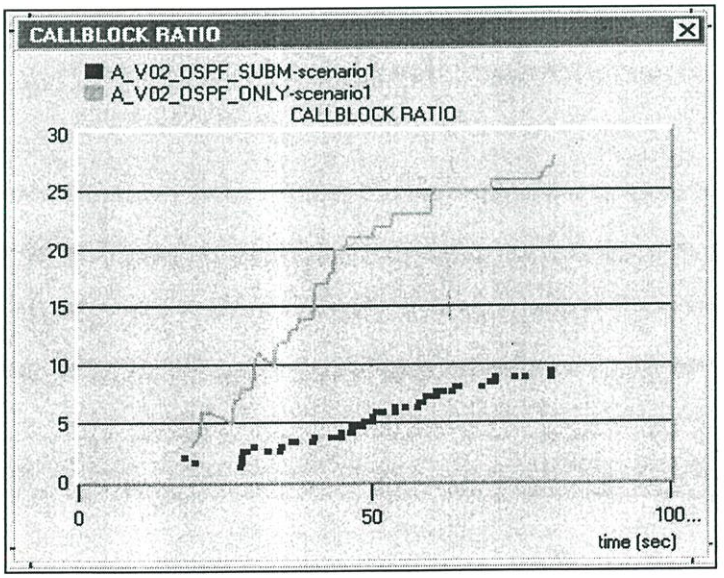
วินาที แล้วตรวจวัดค่าอัตราการปฏิเสธในกระบวนการร้องขอเพื่อเชื่อมโยงเส้นทาง ตรวจสอบขอบเขตของเวลาหน่วง และการสูญหายของแพ็กเกต

โดยวิธีการตรวจวัดอัตราการปฏิเสธการร้องขอเพื่อเชื่อมโยงเส้นทางในระบบนั้นจะเท่ากับอัตราส่วนของจำนวนของการสร้างแพ็กเกตตอบรับแบบปฏิเสธในระบบต่อจำนวนแพ็กเกตร้องขอเพื่อเชื่อมโยงเส้นทางที่ส่งออกมาจากเราท์เตอร์ต้นทางทั้งหมด และในการพิจารณาอัตราการใช้งานช่องสัญญาณเวลาในระบบนั้นจะพิจารณาจากอัตราส่วนของปริมาณช่องสัญญาณเวลาที่มีการจองในระบบต่อจำนวนช่องสัญญาณเวลาทั้งหมดในระบบ หรือจะเรียกอีกอย่างว่าเป็นการวัดประสิทธิภาพการใช้งานทรัพยากรของระบบ

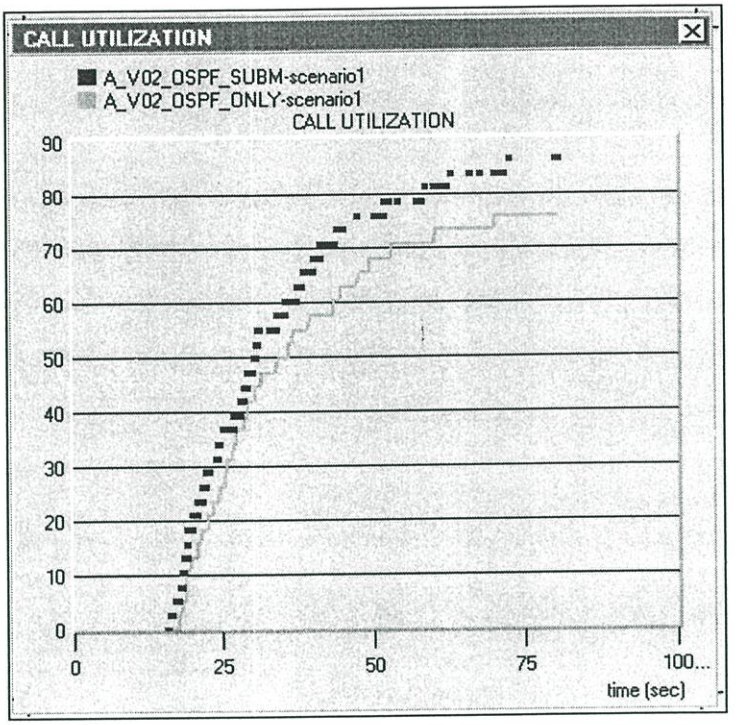
การวัดเวลาหน่วงในกระบวนการสร้างเส้นทางเชื่อมโยงนั้นจะจับเวลาเริ่มส่งแพ็กเกตร้องขอเพื่อเชื่อมโยงเส้นทางจากเราท์เตอร์ต้นทางจนกระทั่งได้รับแพ็กเกตตอบรับแบบตอบรับกลับมาถึงเราท์เตอร์ต้นทางอีกครั้งจึงจะถือว่าการสร้างเส้นทางนั้นประสบความสำเร็จ การจับเวลาจะพิจารณาแยกตามทราฟฟิกแล้วนำมาหาค่าเฉลี่ยของเวลาในการสร้างเส้นทางทั้งระบบในขณะใด ๆ

การวัดอัตราการสูญหายของแพ็กเกตข้อมูลนั้นจะพิจารณาอัตราส่วนของจำนวนแพ็กเกตข้อมูลที่เกิดการสูญหายต่อจำนวนของแพ็กเกตข้อมูลที่ส่งออกมาทั้งหมด

4.2 ผลการจำลอง



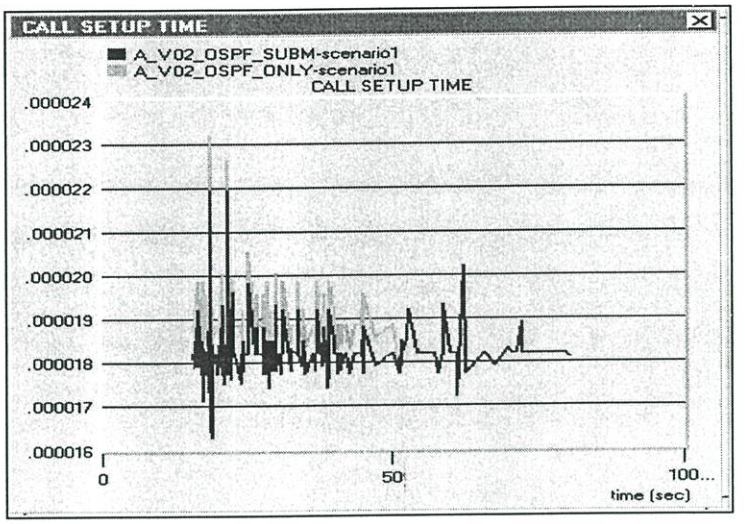
รูปที่ 4.2 เปรียบเทียบอัตราการปฏิเสธการร้องขอเพื่อเชื่อมโยงเส้นทางระหว่างกรณีที่ใช้วิธีการ SUBM และกรณีที่ไม่ได้ใช้วิธีการ SUBM



รูปที่ 4.3 เปรียบเทียบอัตราการใช้งานช่องสัญญาเวลาในระบบระหว่างกรณีที่ใช้วิธีการ SUBM และกรณีที่ไม่ได้ใช้วิธีการ SUBM

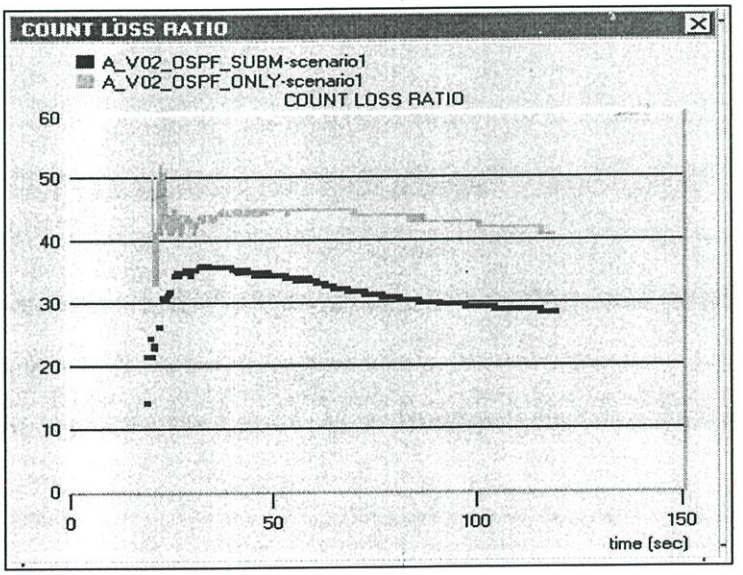
รูปที่ 4.3 จะเป็นการเปรียบเทียบผลการวัดอัตราการปฏิเสธการร้องขอเพื่อเชื่อมโยงเส้นทางระหว่างการหาเส้นทางที่ใช้วิธีการ SUBM และกรณีที่ไม่ได้ใช้วิธีการ SUBM จะเห็นได้ว่ากรณีที่ใช้วิธีการ SUBM นั้นมีอัตราการปฏิเสธการร้องขอเพื่อเชื่อมโยงเส้นทางน้อยกว่าการหาเส้นทางที่ไม่ได้ใช้วิธีการ SUBM ส่วนรูปที่ 4.3 นั้นจะเป็นการเปรียบเทียบผลการวัดอัตราการการใช้งานช่องสัญญาณเวลาในระบบ ซึ่งจะเห็นได้ว่าอัตราการใช้งานช่องสัญญาณเวลาในกรณีที่ใช้วิธีการ SUBM มีอัตราการใช้งานมากกว่าเล็กน้อย ดังนั้นจึงสามารถสรุปได้ว่าในขณะที่อัตราการใช้งานช่องสัญญาณเวลาในระบบใกล้เคียงกันนั้นวิธีการค้นหาเส้นทางที่ใช้การเทียบบิตข้อมูลสถานะการใช้งานช่องสัญญาณเวลามีอัตราการปฏิเสธการร้องขอเพื่อเชื่อมโยงเส้นทางน้อยกว่าวิธีการที่ไม่ได้ใช้การเทียบบิตข้อมูลสถานะการใช้งานช่องสัญญาณเวลา ทั้งนี้เนื่องจากวิธีการที่มีการเทียบบิตข้อมูลสถานะการใช้งานช่องสัญญาณเวลาเป็นการเลือกเส้นทางที่มีช่องสัญญาณเวลาว่างในตำแหน่งที่ซิงโครนัสจึงทำให้มีช่องสัญญาณเวลาว่างที่เหมาะกับการจองจึงช่วยลดอัตราการปฏิเสธการร้องขอเพื่อเชื่อมโยงเส้นทาง

การเปรียบเทียบเวลาหน่วงในกระบวนการสร้างเส้นทางเชื่อมโยงนั้นจะเป็นดังรูปที่ 4.4 ซึ่งจะเห็นว่าการหาเส้นทางที่ใช้วิธีการ SUBM มีเวลาหน่งในกระบวนการสร้างเส้นทางน้อยกว่าการหาเส้นทางที่ไม่ได้ใช้วิธีการ SUBM ทั้งนี้เนื่องจากบางทราฟฟิกถูกปฏิเสธในกระบวนการร้องขอเพื่อเชื่อมโยงเส้นทางจึงต้องมีการคำนวณเส้นทางใหม่และทำการร้องขอเพื่อเชื่อมโยงเส้นทางอีกครั้ง



รูปที่ 4.4 เปรียบเทียบเวลาหน่วงในกระบวนการสร้างเส้นทางเชื่อมโยงระหว่างกรณีที่ใช้วิธีการ SUBM และกรณีที่ไม่ได้ใช้วิธีการ SUBM

ส่วนการเปรียบเทียบผลการวัดอัตราการสูญหายของแพ็กเกตข้อมูลจะเป็นดังรูปที่ 4.5 ซึ่งจะเห็นว่าอัตราการสูญหายของแพ็กเกตในวิธีการค้นหาเส้นทางแบบสั้นที่สุดที่ใช้วิธีการเทียบบิตข้อมูลสถานะการใช้งานช่องสัญญาณเวลามีการสูญหายของแพ็กเกตน้อยกว่าระบบที่ใช้วิธีการค้นหาเส้นทางแบบสั้นที่สุดเพียงอย่างเดียว



รูปที่ 4.5 เปรียบเทียบอัตราการสูญหายของแพ็กเกตระหว่างกรณีที่ใช้วิธีการ SUBM และกรณีที่ไม่ได้ใช้วิธีการ SUBM

4.3 สรุปและวิเคราะห์ผลการจำลอง

จากผลของทั้ง 2 กรณีจะเห็นได้ว่า การนำข้อมูลสถานะการใช้งานช่องสัญญาณเวลามาใช้ร่วมในกระบวนการหาเส้นทางจะทำให้ช่วยลดอัตราการปฏิเสธการร้องขอเพื่อเชื่อมโยงเส้นทางเนื่องจากเส้นทางที่ถูกเลือกมีตำแหน่งช่องสัญญาณเวลาว่างในตำแหน่งที่เหมาะกับการจอง ซึ่งเป็นตำแหน่งที่จะทำให้เกิดการซิงโครนัสกันของการรับ-ส่งแพ็กเกตด้วย ดังนั้นอัตราการถูกปฏิเสธการร้องของจึงน้อยมาก แต่ก็ยังคงมีบางทราฟฟิกที่ถูกปฏิเสธเนื่องจาก บางครั้งแพ็คเก็ตร้องขอการเชื่อมโยงเส้นทางถูกส่งมามากและได้รับเข้ามาเร็วกว่าการปรับค่าของมูลของแพ็กเกตแจ้งสถานะการใช้งานเส้นทางเชื่อมโยง แม้จะมีการประมาณค่าสถานะช่องสัญญาณเวลาที่เปลี่ยนไปของโหนดในเส้นทางหลังจากการเทียบบิตข้อมูลสถานะการใช้งานช่องสัญญาณเวลาแต่ข้อมูลสถานะนี้ก็ยังไม่ใช่ค่าที่เปลี่ยนไปอย่างแท้จริง ดังนั้นบางครั้งการคำนวณของฟังก์ชันค้นหาเส้นทางจึงไม่ใช่ข้อมูลล่าสุดที่แท้จริงทั้งนี้ขึ้นอยู่กับการตั้งค่าคาบเวลาในการสร้างและส่งแพ็กเกตแจ้งสถานะการใช้งานเส้น่ทางเชื่อมโยง ซึ่งการแก้ไขกรณีดังกล่าวนี้สามารถนำไปทำการวิจัยต่อไปได้ ส่วนเวลาหน่วงในการสร้างเส้นทางเชื่อมโยงนั้น การหาเส้นทางที่ใช้วิธีการ SUBM จะมีเวลาหน่วงน้อยกว่าการหาเส้นทางที่ไม่ได้ใช้วิธีการ SUBM เนื่องจากมีการปฏิเสธการร้องขอเส้นทางเชื่อมโยงน้อยกว่านั่นเอง ส่วนเวลาหน่วงในการส่งแพ็กเกตข้อมูลนั้นจะมีขอบเขตที่คงที่ (Bound of Delay) ทั้งสองวิธีเนื่องจากใช้การจัดการบัฟเฟอร์ขาออกที่เราท์เตอร์ทำให้การส่งแพ็กเกตข้อมูลของแต่ละทราฟฟิกถูกส่งออกไปในตำแหน่งช่องสัญญาณเวลาเดิมและได้รับการบริการตรงเวลาจากการใช้วิธีการ W-TDM ซึ่งจะเป็นผลดีต่อการสื่อสารที่มีการจองช่องสัญญาณเวลาแบบตายตัวในเฟรมขนาดคงที่ ในกรณีของการสูญหายของข้อมูลนั้นแม้ว่าจะใช้บัฟเฟอร์ขาออกในเราท์เตอร์มาบังคับอัตราการส่งออกของแพ็กเกตข้อมูลและขยายขนาดของวินโดว์ให้มีขนาดเท่ากับเฟรมเพื่อให้แพ็กเกตสามารถส่งออกจากเราท์เตอร์ได้หมดทุกแพ็กเกต แต่อาจจะเกิดการสูญหายระหว่างส่งข้อมูลไปตามเส้นทางได้ เช่นอาจจะเกิดการชนกันของแพ็กเกตเนื่องจากการเข้ามาพร้อมกันในพอร์ตขาเข้าเดียวกัน แต่จากผลการจำลองจะเห็นได้ว่าระบบที่ใช้การค้นหาเส้นทางแบบสั้นที่สุดร่วมกับการเทียบบิตข้อมูลสถานะการใช้งานช่องสัญญาณเวลามีอัตราการสูญหายของแพ็กเกตน้อยกกว่าวิธีที่ไม่ได้ใช้การเทียบบิตข้อมูล ดังนั้นจะเห็นได้ว่าวิธีการหาเส้นทางโดยนำข้อมูลสถานะการใช้งานช่องสัญญาณเวลามาใช้ร่วมในการพิจารณาเลือกเส้นทางจะทำให้การเลือกเส้นทางที่มีประสิทธิภาพ และได้เส้นทางที่เหมาะสมในระบบการสื่อสารที่มีการกำหนดช่องสัญญาณเวลาแบบตายตัวในเฟรมขนาดคงที่มากกว่าการใช้อัลกอริทึมที่ยังไม่ได้รับการปรับปรุง

บทที่ 5
สรุปและนำเสนอแนวทางในการทำวิจัย

5.1 บทสรุป

การพัฒนาระบบสื่อสารพื้นฐานเพื่อรองรับการให้บริการแบบรับประกันคุณภาพที่หลากหลาย ทำให้เกิดความต้องการวิธีการหาเส้นทางที่ต่างกันไป ในวิทยานิพนธ์ฉบับนี้นำเสนอวิธีการหาเส้นทางสำหรับระบบสื่อสารที่ใช้ในการจองช่องสัญญาณแบบตายตัวในเฟรมขนาดคงที่ โดยนำข้อมูลสถานะการใช้งานช่องสัญญาณเวลามาพิจารณาร่วมกับตัวแปรทราฟฟิกในรูปแบบฟังก์ชันภาระการใช้งานลิงค์และใช้ค่าภาระการใช้งานลิงค์นี้ในการพิจารณาหาเส้นทางที่สั้นที่สุดซึ่งจะเก็บข้อมูลเส้นทางที่เป็นไปได้จำนวนหนึ่งเรียงตามลำดับค่าภาระการใช้งานลิงค์จากน้อยไปมาก แล้วจึงนำเส้นทางที่มีค่าภาระการใช้งานลิงค์น้อยที่สุดมาพิจารณาเทียบบิตเพื่อหาตำแหน่งช่องสัญญาณเวลาว่างที่ซิงโครนัสซึ่งเป็นช่องสัญญาณเวลาที่สามารถจองได้ในเส้นทาง หากพิจารณาเทียบบิตข้อมูลแล้วไม่มีตำแหน่งว่างที่ซิงโครนัสเลยก็จะนำเส้นทางถัดไปมาทำการเทียบบิตอีกครั้ง ทั้งนี้เพื่อลดอัตราการปฏิเสธในกระบวน การร้องขอเพื่อเชื่อมโยงเส้นทางเพราะหากไม่มีช่องสัญญาณเวลาว่างในตำแหน่งที่ซิงโครนัสแล้วจะเกิดการปฏิเสธในกระบวนการร้องขอเพื่อเชื่อมโยงเส้นทางได้ ในขั้นตอนการแลกเปลี่ยนข้อมูลสถานะการใช้งานช่องสัญญาณเวลานั้นจะแนบข้อมูลไปพร้อมกับแพ็กเกตแจ้งสถานะการใช้งานเส้นทางเชื่อมโยง ซึ่งเป็นกระบวนการเดียวกับการแจ้งข้อมูลสถานะการใช้งานเส้นทางเชื่อมโยงในวิธีการหาเส้นทางแบบสั้นที่สุด และในการจำลองได้ใช้โปรโตคอลสื่อสารแบบ W-TDM ซึ่งเป็นโปรโตคอลที่สนับสนุนการสื่อสารแบบที่มีการกำหนดช่องสัญญาณแบบตายตัวในเฟรมขนาดคงที่ โดยนำหลักการ W-TDM มาพัฒนาลงในกระบวนการจัดการแพ็กเกตในขั้นตอนการจัดลำดับแพ็กเกตเพื่อส่งออกในสวิตช์โหนด

การพัฒนาอุปกรณ์จำลองและอัลกอริทึมด้วยโปรแกรมจำลองการทำงานระบบเครือข่าย Opnet ได้ออกแบบการทำงานแล้วสร้างเป็นมอดูลจำลองของอุปกรณ์สื่อสารปลายทาง อุปกรณ์สวิตช์และอุปกรณ์เราท์เตอร์ ซึ่งรายละเอียดอัลกอริทึมและการพัฒนาอัลกอริทึมลงในอุปกรณ์จำลองดังกล่าวจะแสดงในบทที่ 3

ส่วนการจำลองการทำงานในระบบเครือข่ายนั้นได้ใช้ทราฟฟิกที่มีการกระจายแบบคงที่และการกระจายแบบเอกซ์โปเนนเชียลซึ่งจะมีการสุ่มค่าทางเวลาในการสร้างแพ็กเกต ทั้งแพ็กเกตการร้องขอเพื่อเชื่อมโยงเส้นทางและแพ็กเกตข้อมูล รายละเอียดการจำลองจะเป็นดังบทที่ 4 ผลที่ได้จากการจำลองพบว่าอัตราการปฏิเสธการร้องขอเส้นทางเชื่อมโยงในระบบที่ใช้การค้นหาเส้นทางแบบสั้นที่สุดร่วมกับการเทียบข้อมูลบิตสถานะการใช้งานช่องสัญญาณเวลามีค่าน้อยกว่าระบบที่ใช้การค้นหาเส้นทางแบบสั้นที่สุดเพียงอย่างเดียว และเมื่อเปรียบเทียบขอบเขตของเวลาหน่วงและอัตราการสูญหาย

ของข้อมูลจะเห็นว่าในระบบที่ใช้การค้นหาเส้นทางแบบสั้นที่สุดร่วมกับการเทียบข้อมูลบิตสถานะการใช้งานช่องสัญญาณเวลามีประสิทธภาพดีกว่าระบบที่ใช้การค้นหาเส้นทางแบบสั้นที่สุดเพียงอย่างเดียว

5.2 แนวทางในการวิจัยและพัฒนา

เนื่องจากในการจำลองครั้งนี้ได้พัฒนาฟังก์ชันการค้นหาเส้นทางที่เราท์เตอร์ต้นทางเท่านั้น หากเกิดการปฏิเสธในกระบวนการร้องขอเพื่อเชื่อมโยงเส้นทางหรือ หากเกิดปัญหาในเส้นทางเชื่อมโยงจะต้องย้อนกลับมาทำการค้นหาเส้นทางที่เราท์เตอร์ต้นทางเท่านั้น ดังนั้นหากพัฒนาฟังก์ชันการค้นหาเส้นทางที่มีการเทียบบิตข้อมูลสถานะการใช้งานช่องสัญญาณเวลาลงในสวิตช์โหนดด้วยจะทำให้ลดเวลาที่ต้องสูญเสียไปในกระบวนการค้นหาเส้นทางและสร้างเส้นทางใหม่ได้ ทั้งนี้เพราะแพ็กเกตแจ้งสถานะการใช้งานเส้นทางเชื่อมโยงได้ถูกส่งไปยังสวิตช์โหนดทุกตัวดังนั้นข้อมูลจึงถูกส่งไปยังสวิตช์โหนดด้วยซึ่งหากเพิ่มฟังก์ชันการค้นหาเส้นทางเข้าไปสวิตช์โหนดก็สามารถทำงานได้เหมือนเราท์เตอร์และสามารถเพิ่มประสิทธิภาพการใช้งานระบบด้านอื่นต่อไปได้ด้วย โดยเฉพาะกรณีที่มีการเสียหายของลิงค์ที่เชื่อมต่ออยู่ระหว่างโหนดใด ๆ ทำให้ไม่จำเป็นต้องส่งแพ็กเกตกลับไปหาเส้นทางที่เราท์เตอร์ต้นทาง ช่วยให้ลดเวลาที่จะแก้ปัญหากรณีเกิดความเสียหายของลิงค์ดังกล่าว และหากทำให้สามารถทำงานได้อย่างอัตโนมัติแล้ว สวิตชิ่งโหนดในเส้นทางจะเป็นอุปกรณ์ที่มีคุณภาพในการทำงานสูงขึ้น ลดค่าใช้จ่ายและแก้ไขปัญหาได้ในทันที

ในงานวิจัยครั้งนี้ได้กำหนดให้มีคาบเวลาในการส่งแพ็กเกตแจ้งสถานะการใช้งานเส้นทางเชื่อมโยงเป็นแบบคงที่ หากมีทราฟฟิกที่ต้องการร้องขอเพื่อเชื่อมโยงเส้นทางหลายทราฟฟิกจะมีผลต่อการเทียบบิตข้อมูลสถานะการใช้งานช่องสัญญาณเวลาได้ เนื่องจากการข้อมูลในระบบมีการเปลี่ยนแปลงแต่ยังไม่มีการแจ้งข้อมูลใหม่ ทำให้เกิดความคลาดเคลื่อนในการเทียบบิตเพื่อเลือกเส้นทางได้ ดังนั้นวิธีการที่ควรจะทำต่อไปคือการพัฒนาให้ช่วงเวลาในการสร้างและส่งแพ็กเกตแจ้งสถานะการใช้งานเส้นทางเชื่อมโยงมีการเปลี่ยนแปลง ปรับเปลี่ยนได้ตามปริมาณความต้องการใช้งานช่องสัญญาณเวลา หากมีอัตราการร้องขอเพื่อใช้งานเส้นทางเชื่อมโยงมากขึ้น ก็สามารถปรับคาบเวลาในการส่งแพ็กเกตแจ้งสถานะการใช้งานเชื่อมโยงได้เร็วขึ้น แต่หากมีอัตราความต้องการใช้งานเส้นทางเชื่อมโยงน้อยลงก็เพิ่มระยะห่างของคาบเวลาในการสร้างและส่งแพ็กเกตแจ้งสถานะการใช้งานเส้นทางเชื่อมโยงได้เพื่อลดปริมาณทราฟฟิกในระบบนั่นเอง

เอกสารอ้างอิง

[1] Paul F. and Geoff H. **Quality of Service: Delivering QoS on the Internet and in corporate networks.** New York : John Wiley, 1998.

[2] Heinrich M. and Gerd A. **Synchronization in digital communications.** New York : John Wiley, 1990.

[3] George C. and Christopher Y. **ATM and multi - protocol networking.** New York : McGraw-Hill, 1997.

[4] Kamiyama N., Ohta C. and Toda H. "Quasi-STM Transmission Method based on ATM Network." **IEEE Global Telecommunication Conference**, Vol.3, 1994, Pp. 1808-1814.

[5] Anna H. **Multimedia applications support for wireless ATM networks.** Upper Saddle River, New Jersey : Prentice Hall, 2000.

[6] Natalie G. and Sudhakar G. **Quality of Service in ATM network: state-of-the-art traffic management.** New Jersey : Prentice Hall PTR, 1999.

[7] Awduche D., Malcolm J., Agogbua J., O'Dell M. and McManus J. **Requirements for Traffic Engineering over MPLS.** [Online]. Available : http://www.faqs.org/rfcs/rfc2702.html.

[8] Eric S. **QOSPF: Quality of Service Extensions to OSFP or Quality of Service Path.** [Online]. Available : http://mirror.switch.ch/ftp/doc/ietf/96jun/ qosr-slides1-96jun .

[9] Mayuree L. and Rattikorn V. "QoS guarantee in ATM network employing Window based TDM-like Scheduling." **APSBC2000**, 2000, Pp. 383-388.

[10] Mike G. **Private Network - Network Interface Specification Version 1.0.** [Online]. Available : ftp://ftp.atmforum.com/pub/approved-specs/afpnni-0055.000.

[11] David E. and Darren L. **ATM Theory and Application.** McGraw-Hill, 1999.

[12] Golestani S. "A framing strategy for congestion management." **IEEE Journal on Selectes Areas in Communications**, Vol. 9, 1991, Pp. 1064-1077.

[13] Richard W. and Karsten S. "Dynamic Window-Constrained Scheduling for Multimedia Applications." **EEE International Conference on Multimedia Computing and Systems**, Vol. 2, June, 1999.

[14] Atsushi I. and Norihito F. "A Hierarchical Multilayer QoS Routing System with Dynamic SLA Management." **IEEE Journal on selected areas in communication**, Vol.18, No.12, December 2000, Pp. 2603-2616.

[15] Hwa-Chun L. and Ping-Chin F. "Finding Available Bandwidth in Multihop Mobile Wireless Networks." **Proceedings of the IEEE Vehicular Technology Conference**, Tokyo, Japan, May, 2000.

[16] Hwa-Chun L. "QoS Routing in Ad Hoc Wireless Networks." **Proceedings of the 23rd Conference on Local Computer Networks**, 1998, Pp. 31-40.

[17] Phongkaew K., Chotipat P., Rattikorn V., Chakraborty G. and Shiratori N. "On Precomputing of QoS Paths." **In IEEE International Workshop on Intelligent Signal Processing and Communication Systems (ISPACS'99)**, Phuket, Thailand, December 1999, Pp. 17-20.

[18] Katevenis M., Sidilopoulos S. and Couracoubetis C. "Weighted Round-Robin cell multiplexing in general purpose ATM switch chip." **IEEE Journal on Selected Areas in Communication**, vol.9, October 1991, Pp. 1265-1279.

[19] Keith W. Ross and James F. **Computer Networking: A Top-Down Approach Featuring the Internet.** New York : Addison Wesley Longman, Inc. 2001.

[20] Rattanawit S., Lertwatechakul M. and Sooraksa P. "The Delay-Bounded Source Model." Proceedings of the International Conference on Scientific & Engineering Computation, 2002, pp. 340-343.

[21] Kaewyongphang P., Pornavalai C., Varakulsiripunth R., Chakraborty G. and Shiratori. "On Precomputing of QoS Paths." In IEEE International Workshop on Intelligent Signal Processing and Communication Systems (ISPACS'99), Phuket, Thailand, December 1999.pp. 17-20.

[22] สัลยุทธ์ สว่างวรรณ. **เครือข่ายคอมพิวเตอร์**. กรุงเทพมหานคร : บริษัท เพียร์สัน เอ็ดดูเคชั่น อินโดไชน่า จำกัด. 2542.

[23] Cisco Systems, et.al. **Internetworking Technologies Handbook**, Third Edition. United States of America: Cisco Press logo is a trademark of Cisco Systems, Inc. 2001.

[24] Martin de Prycker. **Asynchronous Transfer Mode Solution for Broadband ISDN.** Third edition. Prentice Hall International (UK) Limited. 1995.

[25] Halsall, Fred. **Data Communications, Computer Networks and Open Systems.** Third edition, United State of America, Addison Wesley Publishing Company, Icn. 1992.

[26] Panagiotis N. Zarrow, Myung J. Lee, Tarek N. Saadawi. "Interparticipant Synchronization in Real-Time Multimedia Conferencing Using Feedback." IEEE/ACM Transactions on networking, Vol.4, No. 2, April 1996, pp. 173-180.

[27] Wha Sook Jeon and Dong Geun Jeong. "Comparison of Time Slot Allocation Strategies for CDMA/TDD Systems." IEEE Journal on selected areas in communication, Vol.18, No.7, July 2000, pp. 1271-1277.

[28] Casetti C., Cigno R.Lo, Mellia M., Munafo M. "A New Class of QoS Routing Strategies Based on Network Graph Reduction." The International Journal of Computer and Telecommunications Networking, Vol. 41, Issue 4 March 2003, pp. 475 – 487.

[29] Cheng T. H., Chen J. "Performance of fast bandwidth reservation with multipath routing." IEEE Proc of Commun. Vol. 145, No. 2, April. 1998, pp.80-86.

[30] Minseok Song, Naehyuck Chang, and Heonshik Shin. "A New Queue Discipline for Various Delay and Jitter Requirements in Real-Time Packet-Switched Networks." Seventh International Conference on Real-Time Systems and Applications (RTC SA '00) Cheju Island, South Korea, December 12 -14, 2000, pp.191-198.

[31] Roch A. Guérin, Ariel Orda. "Networks with advance reservations: The routing perspective." IEEE INFOCOM 2000 - The Conference on Computer Communications, no. 1, March 2000, pp. 118-127.

[32] Donna Ghosh, Venkatesh Sarangan, and Raj Acharya. "Quality of Service Routing in IP networks." IEEE Transactions on Multimedia - Special issue on Multimedia over IP, June 2001.

[33] Gangadharan Kannan. "Selective Flooding for Improved Quality-of-Service Routing." M.S. Thesis, Computer Science Department, Worcester Polytechnic Institute, Spring 2000.

[34] Aarthi, V. and Siromoney, A. "Quality of Service Routing in DSR," Trusted Internet Workshop 2002, International Conference on High Performance Computing (HiPC 2002), Bangalore, India, Dec 18, 2002.

ภาคผนวก

ภาคผนวก ก.
บทความที่ได้รับการตีพิมพ์ในวิทยานิพนธ์

[1] Ajchara K., Supannee R., Mayuree L. and Pitikate S. "The Effect of Unsynchronized Time Slot Allocation on W-TDM Network." (ISCIT 2003), September 3-5, Pp. 656-659.

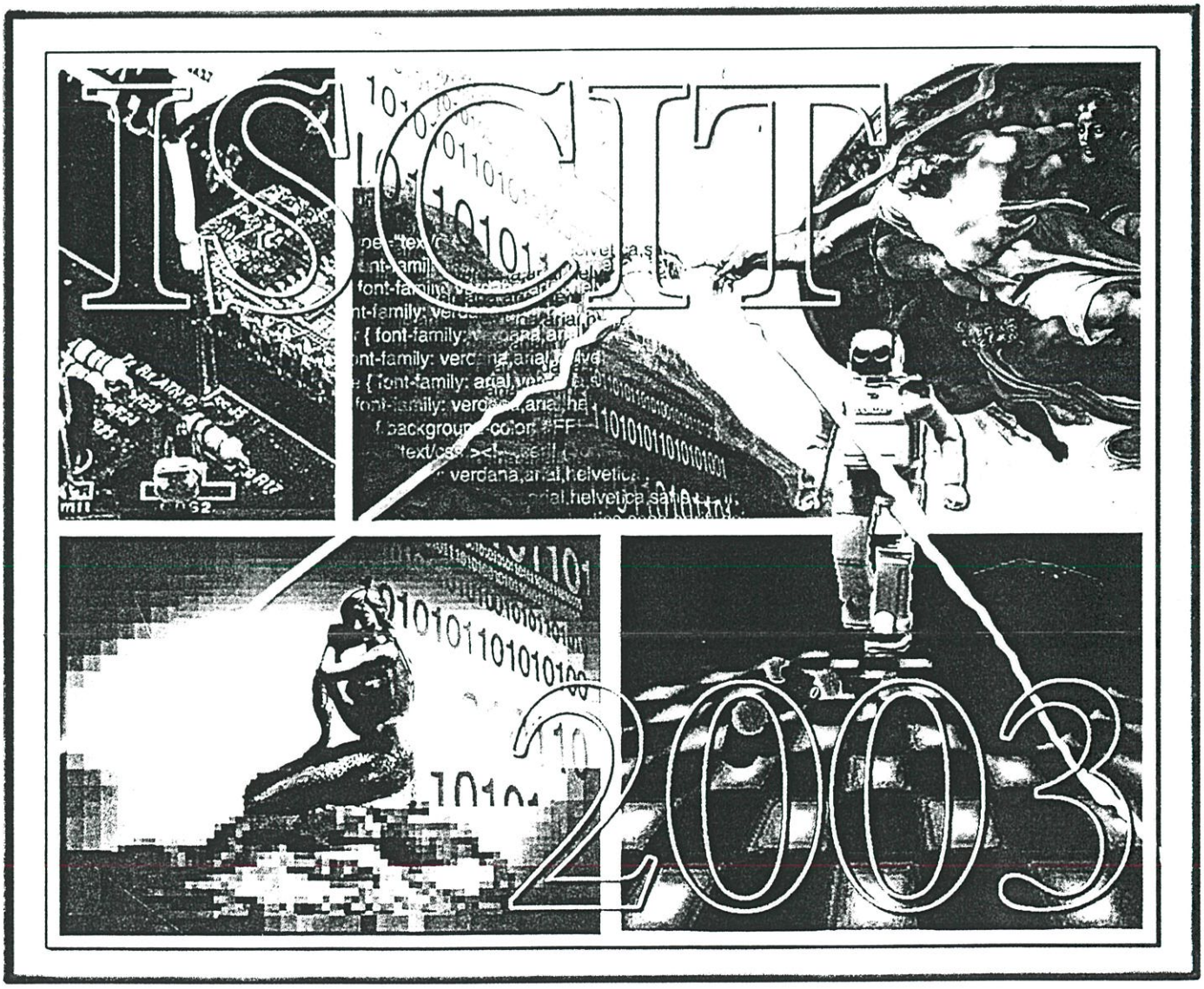
Volume II

Proceedings

The Third International Symposium on Communications and Information Technologies

September 3-5, 2003

BP Samila Beach Hotel and Resort, Songkhla, Thailand



ISCIT 2003

ISBN 974-644-437-9

The Effect of Unsynchronized Time Slot Allocation on W-TDM Network

A. Kaewthep, S. Rattanawit, M. Lertwatechakul, P. Sooraksa

Department of Information Engineering, Faculty of Engineering,
King Mongkut's Institute of Technology Ladkrabang, Bangkok 10520, Thailand.
Email: s3061101@kmitl.ac.th, s3061093@kmitl.ac.th, klmayure@kmitl.ac.th.

ABSTRACT

Window based TDM (W-TDM) is a new scheduling concept that was proposed to enhance today data transmission system. The objective of the model is to provide QoS service for both of synchronous and semi-synchronous traffic. The scheduling strategy is using service window concept to specify acceptable time period of any traffic flows within each time frame. By applying the method, a traffic flow must regulate its traffic to the defined pattern in order to achieve a committed service level, or it could experience packet lost. To investigate the W-TDM performance and its effect on traffic flows of which behavior are different, we have implemented the W-TDM concept in a simulation model. A simulation scenario was set up and the simulation results were analyzed. The results of simulation accepts the performance of W-TDM concept.

KEYWORDS

scheduling algorithm, service window, frame-based, semi-synchronous transfer mode, Opnet simulation

1. INTRODUCTION

Scheduling is a necessary mechanism that was used to manage packet handling in switching node. Scheduling task is sorting of packet order to provide difference QoS guarantees for each traffic. W-TDM is a scheduling method that was proposed to use as a transmission network between synchronous systems and semi-synchronous systems. Data transmission in synchronous system such as TDM and STM, transmission device has to buffer and cut data into the same size pieces and transmit them exactly within the assigned time slots. The allocated time slots could not be used by the other traffic flows even if the allocating traffic flow has nothing to be sent. This discipline induces low utilization in synchronous network. An example of semi-synchronous scheduling is Q-STM [2], the scheduling provides guarantee slots for static STM traffics and best effort slots for ATM traffics. In Q-STM, STM traffics are treated as a premium class of service, ATM traffics can use only time slots that are unused by STM traffics. Normally, scheduler of semi-synchronous transmission network should support for differentiated services.

By using W-TDM method, an early arrival packet may be buffered to wait for its service time, or it maybe dropped in case that it come too early. This concept gives more flexibility on arrival time compare to TDM method. As same as Q-STM, the unused slots of every W-TDM frame could be shared by extra-packets of guaranteed flows and or best-effort traffic flows' packets. Excluding to increasing flexibility of service time, one importance objective of the service window was decreasing complexity of designing stage. Moreover, the W-TDM supports multi-class of service by using enhanced buffer management strategy.

To prove the W-TDM performance and study on the effect of W-TDM window size, we have implemented the W-TDM concept into a simulation model. The model was designed and simulated onto the Opnet 6.0 simulation tool. In this paper, we show how to develop the W-TDM model, describe the defined scenario and its simulation results. The simulation results were analyzed to make the conclusions of the studying develop the W-TDM model, describe the defined scenario and its simulation results. The simulation results were analyzed to make the conclusions of the study.

2. W-TDM CONCEPT

W-TDM is a scheduling scheme that was proposed to provide differentiated services in a data communication network. The W-TDM scheduling utilizes frame-based concept and service window strategy. The frame-based and fixed time slot allocation concept is used to provide synchronous and semi-synchronous control when the service window strategy was introduced to limit available time period of a system for every traffic.

Frame-based transmission method is used to define a time frame structure that consists of a number of time slots. In order to ease the allocation process, the W-TDM uses fixed size time slot onto a fixed frame size. Since number of time slot is fixed and equally in every frame, size of a time slot could be referred as a minimum bandwidth that the system could provide to the users. The W-TDM services a traffic flow by determining its allocated time slot position according to the current time frame. To achieve the committed QoS level, a traffic flow has to transmit its packet within an available period determined by the service window.

Service window concept is a new discipline that was proposed to use in W-TDM. The concept is defined to allow more flexible arrival time compare to synchronous networks, and in order to restrict traffic behavior compare to asynchronous networks. Moreover, service window concept also introduces more advantage of decrease the complexity in switch designing stage and reduces required buffer quantity. Service window could be implemented by just using a more strictly buffer management concept.

Fig 1.1 shows service window concept. In the example, the actual time is t=3, service window scope is covering the 3rd slot to the 7th slot and this is the expected arrival time of the traffic of 3rd slot. That means a packet

would be accepted by the switch if and only if the packet is belong to the traffic flows that allocated time slot number 3rd or 4th or 5th or 6th or 7th or would be dropped in the other case. At this time a packet from traffic that reserved on the 5th slot arrived. Althrough packets of 5th slot should be received at time t=5 but now traffic description of 5th slot in translation table is determining by the service window so that the packet would be accepted. On the other hand, if the packet of 8th slot and a packet of the 1st slot come at this time, the both packets would be dropped because their traffic description could not be seen by the service window.

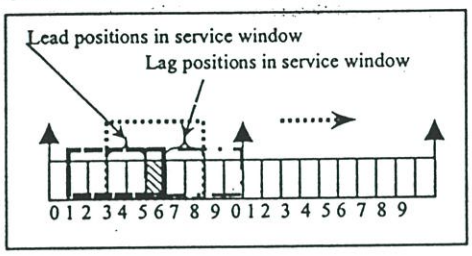


Fig. 1.1 Lead-lag consideration

In order to provide differentiated services, buffers in W-TDM is classified into 3 groups, guarantee buffers, bursty buffers and best effort buffers. The guarantee buffers are provided for constant traffic such as TDM and CBR. The bursty buffers are provided for bursty traffic like VBR and ABR in ATM. [1] Best effort buffer is provide for UBR and the other traffics.

3. DESIGN AND IMPLEMENTATION

One of the convenient ways to investigate performance of a communication protocol is simulation. We used The Opnet 6.0 as a network simulation tool to implement and study the W-TDM concept. To develop a W-TDM switch model, we divided functions of the switch into 3 functions: call set-up, routing and scheduling. Each function was implemented as a process model and they can communicate through messaging mechanisms.

The W-TDM concept was implemented in the scheduling process of a switching node model. The process consists of scheduling and forwarding subtask that was invoked by a set of specific events. By using Opnet 6.0, the scheduling process was implemented as a state transition diagram that consists of 5 states as was shown in Fig 1.2. Tasks of each state are described as below

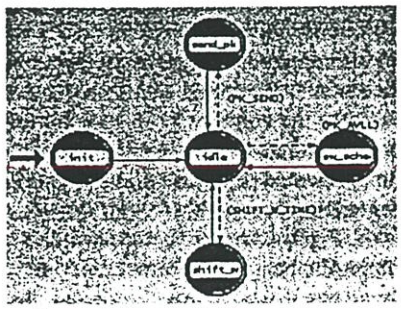


Fig. 1.2 W-TDM scheduling process

- **init state** is the first state of the scheduling process. Its task is to set initial value of every state variable and get parameters from external configuration that could be specified by user at modeling time and or at simulating time. We also provided external interface for routing configuration of a W-TDM switch by defining an external text file format. An initial routing information could be set-up and import into the system, this could help user to consider behavior of a system in the interesting environment.

- **shift_w state** is a state that was defined to shift the service window to determine the next set of acceptable time slots of the system. The next set of time slots consists of every time slots of the previous service window excluding for the oldest one and include the next time slot. To do this, we have to implement the service window concept as a fixed size window that was moved in circular fashion onto a translation table. The first slot of the new frame will be reached again and again when the service window was moved beyond the last slot of the current frame.
- **sw_schedule state** is the scheduling state of W-TDM. While receiving a packet, VCI value (Virtual Channel Identifier) will be retrieved from the received packet header. The VCI value is used as a key to search for a corresponding output port and a VCI value to be used in translation process. The matching process was done on a set of tuples of the translation table which is under determining scope of the current service window. If VCI values are matched, the header of packet would be translated, VCI value would be changed and the packet would be sent to the according position of forwarding buffer in the specified outgoing port. Finally the packet could be transmitted exactly in its allocated time position. In case that VCI values are not matched, the packet would be dropped at this switching node.
- **send_pk state** acts as a forwarding process to transmit packets one by one from the forwarding buffers to the outgoing port associated to the service time of each buffer. The forwarding buffers are used to store eligible packets before to be transmitted to the next node. Forwarding buffer size is defined to have the same size as the service window. Forwarder shifts its buffer pointer every unit of time that equal to the shifting time of service window in circular manner. If a packet is received exactly on its expected arrival time, it will be transmitted without waiting in the forwarder buffer. On contrary, if a packet's arrival time is fell between its earliest arrival time and its expected arrival time those corresponding to its allocated time slot and location of the service window, the packet would be stored and waited for its service time. In case that a packet comes too early or too lately (after its expected arrival time) the packet would be dropped.

4. SIMULATION

We used 3 types of source model to investigate how W-TDM takes effect to different kinds of source model. The first source type generates constant traffic while the second source type generates delay-bounded traffic [4] that its packet generating time is lead or lag to the expected packet-arrival time under a limited time

bounded, the last source type generates traffic in distributed inter-arrival PDF pattern. In this experiment we have set service window size equal to 5 and the frame size 10 to every switch. Fig. 1.3 shows the W-TDM network model. The scenario was configured as followed.

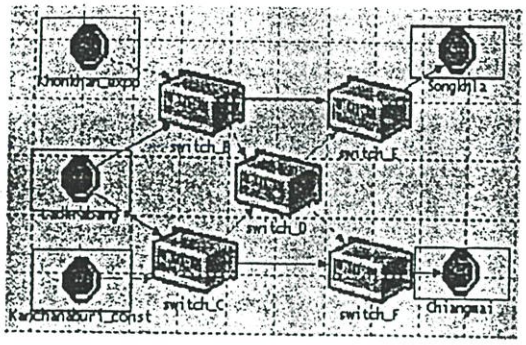


Fig. 1.3 W-TDM network model

Kanchanaburi_const: All sources from this subnet sent traffics to the Songkhla subnet through switch_C, switch_D and switch_E. Structure of the subnet is shown in Fig 1.4. Source Src0, Src1, Src2, Src3 and Src4 were generating traffic with constant packet size every a constant interval.

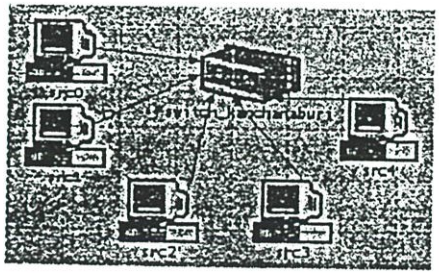


Fig. 1.4 Structure of Kanchanaburi_const subnet

Khonkhan_expo: The subnet is shown in Fig 1.5, Khonkhan_expo subnet consisted of 5 exponential delay-bounded sources and 5 inter-arrival distributed pdf sources.

- The inter-arrival distributed pdf traffics are generated by constant function, exponential function, poisson function, bernoulli function and uniform function. All of these traffics were transmitted to the Server_P.
- The 5 delay-bounded sources were generating the traffics with delay-variance of exponential function, the variance was set to 0.5 at src_dbs_expo_0, set to 1 at src_dbs_expo_1, set to 2 at src_dbs_expo_2, set to 3 at src_dbs_expo_3 and set to 5 at src_dbs_expo_4. These 5 sources were sending traffics to Chiangmai on the path through switch_B, switch_D and switch_F.

Ladkrabang: In Fig 1.6, Src0, Src1, Src2, Src3, Src4 of subnet Ladkrabang were set to transmit traffics to Chiangmai through switch_C and switch_F. While Src5, Src6, Src7, Src8 and Src9 of Ladkrabang subnet were set to transmit traffics to Songkhla through switch_B and switch_E path. The sources in Ladkrabang subnet were used to investigate the effect of lead-lag generated traffics. The lead-lag traffic were transmitted into unsynchronized time slots which leaded or lagged to the reserved time slots under upper-bound of delay-variation. Table_1 show lead-lag position of received-time of these source and result of the packets transmission.

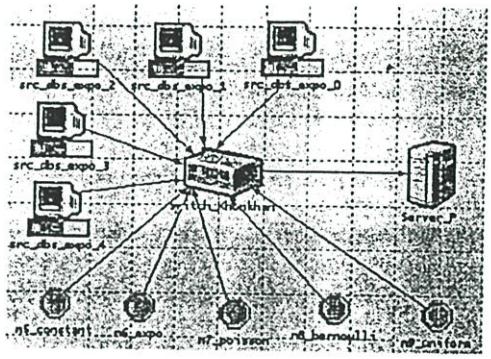


Fig. 1.5 Structure of Khonkhan_expo subnet

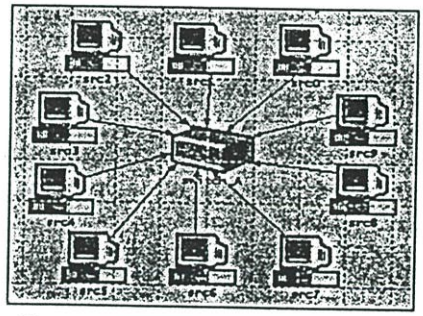


Fig. 1.6 Structure of Ladkrabang subnet

5. SIMULATION RESULT

The simulation result shows that a traffic could achieve lossless and non-delay jitter if its packets were regulated in W-TDM specified- manner.

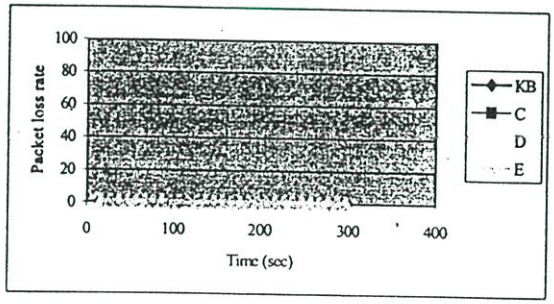


Fig. 1.7 Lost rate of constant traffics of sources in Kanchanaburi_const subnet

- The Fig 1.7 shows the summation of packet loss rate of constant source traffics in Kanchanaburi_const subnet at switch C, switch D and switch E. All packets of constant traffic were transferred successfully with constant transit delay.
- Fig 1.8 shows that just some packets of src_dbs_expo_

4 which were sent at the beginning of the simulation were lost when the other flows were exeperienced no lost. The result proved that the delay-bounded source traffics which were set to have small deviation than the size of window would totally be accepted. On the other hand, some packets from the delay-bounded source traffic with wide deviation delay time could be dropped in case that they arrived too early or too lately.

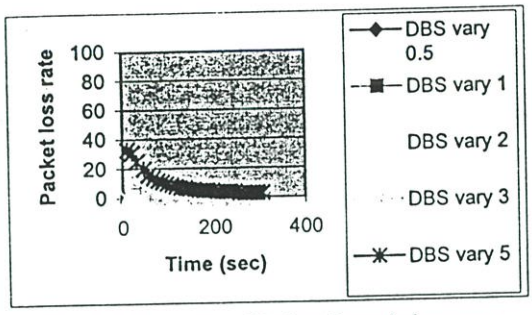


Fig. 1.8 Lost rate of Delay-Bounded sources

- Packets of inter-arrival distributed pdf sources are analyzed related to its distribution pattern. Packets of node n5_constant have not lost and experienced no delay jitter. Thus, the synchronous traffics achieved their promised service in W-TDM network. The result also shown that packets of node n7_poisson were lost on the average 46.548% while some packets of node n6_expo were lost 51.24%. Packets of node n8_benoulli were lost 90% and packets of node n9_uniform were lost with 66.207% loss rate is shown in Fig. 1.9. Some packets were lost because of their arrival-time were out of service window. Thus, the lost-rate of these sources depends on deviation factor of each distribution function. So type of distribution function and its deviation factor are an effective factor in W-TDM network.
- Trafficss from Ladkrabang subnet were classified into 3 groups. The first group was a type of traffic which was received at service time and src0 and src3 were configured to generate this type of traffic. The second group was a type of traffic that were received at leading position to their expected arrival time, and this type of traffic was set to the src2, src5, src8 and src9. The third group was a type of traffic that were received at leading or lagging position that was out of service window scope. This type of traffic were generated from source src1, src4, src6 and src7.

Table 1. The effect of lead-lag position

The order of position array in Translation Table	The order of packet received	Lead-Lag of service position in window	Transfer to next switching node
Src 0	0	0	Success
Src 1	6	Lag 1	Lost
Src 2	9	Lead 4	Success
Src 3	3	0	Success
Src 4	8	Lag 2	Lost
Src 5	2	Lead 3	Success
Src 6	7	Lag 1	Lost
Src7	1	Lag 2	Lost
Src 8	4	Lead 3	Success
Src9	5	Lead 2	Success

The results from Table1. and fig. 1.10 show that all packets of src1, src4, src6 and src7 were lost but all packets from the others were 100% transmitted successfully.

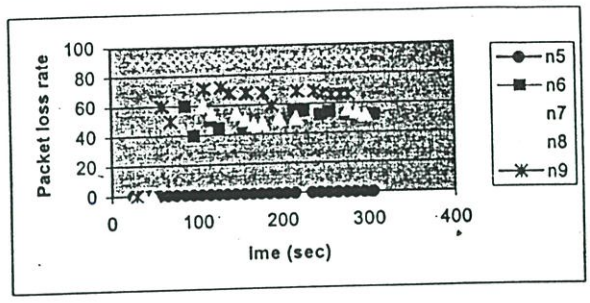


Fig. 1.9 Lost rate of distribution function

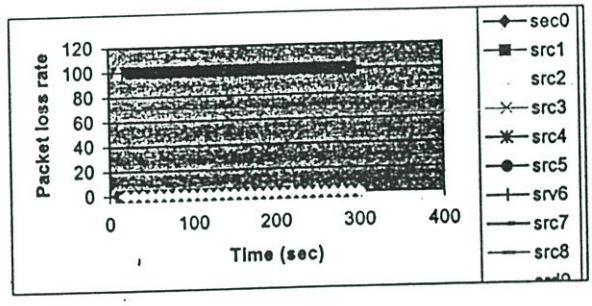


Fig. 1.10 Lost rate of delay constant sources

6. CONCLUSIONS

The service window concept of W-TDM scheduler increases flexibility of service time period while provides guaranteed service on transit delay-bound, delay jitter and loss rate. Synchronous traffic and delay-bounded traffic can fully achieve W-TDM QoS if and only if their traffics were regulated to be synchronized to the service window under the system defined pattern. On the other hand, some or all packets of traffic flows those are received when its time slot could not be determined by the service window would not be serviced. That means data communication in W-TDM is flexible related to the width or service window and a traffic flow has to obey the rule to avoid packet lost.

The next thing we have to do is to study and develop an appropriate routing algorithm to be used in fixed slot allocation network for synchronous and semi-synchronous traffics. And then study on behavior of real-application traffic type such as VoD, IP-telephony and so on. The real traffic behavior would be model and use as traffic sources of the next simulation model.

REFERENCES

[1] David E. McDysan, Darren L. Społ and Application, Sig. Edition, McGraw-Hill, 1999
[2] N. Kamiyama, C. Ohta and H. Toda: "Quasi-STM Transmission Method based on ATM Network," IEEE Global Telecommunication Conference, Vol.3, 1994.
[3] M. Lertwatechakul and R. Varakulsiripunth: "QoS guarantee in ATM network employing Window - based TDM – like Scheduling", APSBC2000, pp. 383-388, 2000.

ประวัติผู้เขียน

นางสาวอัจฉรา แก้วเทพ เกิดเมื่อวันที่ 1 สิงหาคม 2519 ที่จังหวัดหนองคาย สำเร็จการศึกษาระดับปริญญาตรีในหลักสูตรวิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมโทรคมนาคม จากมหาวิทยาลัยเทคโนโลยีสุรนารี ปีการศึกษา 2541 และเข้าศึกษาในระดับปริญญาโทในหลักสูตรวิศวกรรมศาสตรมหาบัณฑิต สาขาวิศวกรรมไฟฟ้า ภาควิชาวิศวกรรมสารสนเทศ คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง เมื่อปีการศึกษา 2543

ผลงานทางวิชาการที่ได้รับการตีพิมพ์นอกเหนือจากวิทยานิพนธ์

[1] Supannee R., Ajchara K., Mayuree L. and Pitikate S. "The Optimum Service-window Size of W-TDM Network." The 4th Information and Computer Engineering Postgraduate Workshop, 22-26 January 2004, pp 203-208.